



BACHELORSTUDIENGANG GEODÄSIE UND GEOINFORMATION

## BACHELORARBEIT

von

Laura Jensen

## AUTOMATISCHE DETEKTION VON BOMBENTRICHTERN

Bonn 2008

BETREUER:

PROF. DR.-ING. WOLFGANG FÖRSTNER

DIPL.-INFORM. MARTIN DRAUSCHKE

RHEINISCHE FRIEDRICH-WILHELMS-UNIVERSITÄT BONN

INSTITUT FÜR GEODÄSIE UND GEOINFORMATION

PROFESSUR FÜR PHOTOGRAMMETRIE



## **Bacheloraufgabe**

für Laura Jensen

### **Automatische Detektion von Bombentrichtern**

Während des 2. Weltkriegs wurden mehrere Millionen Luftbilder über Deutschland aufgenommen. Heute unterstützen diese Bilder u. a. die Arbeit des Kampfmittelbeseitigungsdienstes.

Aufgabe der zu vergebenden Bachelorarbeit ist es, in den digitalisierten Luftbildern Bombentrichter automatisch zu erkennen und zu lokalisieren. Wegen der vielfältigen Beschaffenheit des Geländes und der unterschiedlichen Beleuchtungsverhältnisse, muss ein komplexes Bildmodell der Bombentrichter entwickelt werden, in dem auch der Sonnenstand bzw. der Schatten des Objekts aus der ungefähren Aufnahmezeit berücksichtigt werden. Zunächst werden verschiedene Matching-Verfahren bzw. Musterlernverfahren auf ihre Verwendbarkeit für die Detektion von Bombentrichtern im ländlichen Raum evaluiert.

Das implementierte Verfahren soll anschließend auch daraufhin untersucht werden, inwiefern es für die Detektion von Bombentrichtern in städtischen Bereichen eingesetzt werden kann.

Prof. Dr.-Ing. Wolfgang Förstner

Betreuer:	Prof. Dr.-Ing. W. Förstner, Dipl.-Inform. M. Drauschke
Ausgegeben am:	29.02.2008
Abgabetermin:	31.07.2008
Abgegeben am:	16.07.2008

## ERKLÄRUNG

Hiermit erkläre ich eidesstattlich, dass ich die vorgelegte Arbeit selbständig und ohne unzulässige Hilfe angefertigt habe. Es wurden keine anderen als die angegebenen Quellen und Hilfsmittel verwendet.

Bonn, den 16. Juli 2008

Laura Jensen



## ZUSAMMENFASSUNG

Der Kampfmittelbeseitigungsdienst der Bezirksregierung Arnsberg nutzt Luftbilder aus dem Zweiten Weltkrieg zur Detektion von Blindgängern. Aufgrund der großen Anzahl an Bildern (über 300000) ist die Suche sehr aufwändig. Die Arbeit der Auswerter könnte erleichtert werden, wenn sie eine Karte hätten, auf der die Dichte der Bombardierung dargestellt ist. Um diese Karte zu erstellen, ist ein Verfahren notwendig, das die Bombentrichter auf den Bildern automatisch detektiert. Dieses wurde in der vorliegenden Bachelorarbeit realisiert.

Da die Trichter sich in ihrer Gestalt und Größe stark unterscheiden, muss ein Ansatz zur Detektion gewählt werden, der mit diesen Variationen umgehen kann. Der Algorithmus führt eine Kandidatensuche mittels Kreuzkorrelation des Bildes mit einem repräsentativen Trichter-Template in verschiedenen Größen durch und klassifiziert die gefundenen Kandidaten anschließend. Die Klassifizierung erfolgt mit Hilfe der Wahrscheinlichkeitsdichte der Verteilungen der Klassen Trichter und Hintergrund. Um die Verteilungsparameter zu schätzen, ist die Dimensionsreduktion des Merkmalsraums der Trainingsdaten mit einer Hauptkomponentenanalyse (PCA) und einer linearen Diskriminanzanalyse nach Fisher (LDA) und anschließender Projektion in den Unterraum notwendig. In dieser Arbeit wurde das Verfahren mit einer Trichterklasse implementiert, es kann aber gut auf verschiedene Trichterklassen erweitert werden.

Der Algorithmus zur Bombentrichterdetektion wurde in Matlab implementiert. Nach der Vorverarbeitung des Bildmaterials mussten zur Erstellung des Templates zunächst Trainingsbilder annotiert werden. Außerdem waren bei der Umsetzung verschiedene Parameter, wie z.B. die Templategrößen zur Kandidatensuche, die Dimension des PCA-Raums und die Bildausschnittsgröße bei der Klassifikation zu bestimmen.

Zur Beurteilung der Ergebnisse wurde der Algorithmus auf den Trainingsbildern getestet und die Ergebnisse mit den Referenzdaten verglichen. Je nachdem ob vier oder fünf Templategrößen verwendet werden, können mit dem erstellten Template etwa 75% oder 80% der Trichter erfasst werden. Nach der Klassifikation werden mit dem implementierten Algorithmus je nach Konfiguration zwischen 70% und 64% der Trichter detektiert, dabei ist die Relevanz allerdings sehr gering. Maximal sind etwa 31% der als Trichter klassifizierten Bildausschnitte auch tatsächlich Bombentrichter.

Bei der Analyse der false positives auf Testbildern ergab sich, dass bestimmte Bildstrukturen, wie Hausdächer, Schattenwurf an Straßen, Texturen in Feldern oder Waldstrukturen immer wieder fälschlicherweise als Trichter klassifiziert werden. Bei der Untersuchung der nicht detektierten Bombentrichter konnten Trichterklassen abgeleitet werden, die mit dem erstellten Template nicht detektiert werden. Mit den Testbildern wurde außerdem die Möglichkeit untersucht, die Bilder mit Hilfe der Bombentrichterdetektion in die Kategorien schwache, mittlere und starke Bombardierung einzuordnen. Hierbei wurden 73% der Bilder der richtigen Kategorie zugeordnet. Bei einer Steigerung der Relevanz und der Annotation weiterer Testbilder ist eine bessere Einordnung zu erwarten.

Insgesamt liegt mit dieser Arbeit ein vielversprechender Ansatz zur Bombentrichterdetektion mit großer Erweiterungsmöglichkeit vor.

## ABSTRACT

The explosive ordnance disposal service of the district government Arnsberg uses aerial photographs from World War II for the detection of duds. Because of the great number of images (over 300000) this search is very time-consuming. It would be of help to the specialists analyzing the photographs if there existed a map showing the density of bombing in a particular region. To develop such a map, a procedure is needed which detects bomb craters on images automatically. This has been done in the present bachelor thesis.

Since the craters differ significantly in appearance and size, we have to find an approach which is able to deal with those variations. The algorithm executes a candidate search by performing a cross correlation of the image with a representative crater-template in different sizes. Afterwards it classifies the candidates found. The classification is done by means of probability density of the distributions of the classes crater and background. To estimate the distribution parameters, the dimension of the feature space of the training data has to be reduced using principal components analysis (PCA) and Fisher's linear discriminant analysis (LDA) and projection into subspace. In this thesis the procedure is realised with only one class for craters, but it can easily be extended to more.

The algorithm for bomb crater detection was implemented in Matlab. After pre-processing the image material craters were annotated on some training images. This was necessary for creating the template. Several parameters had to be defined during the realisation, e.g. the sizes of the template used for candidate search, dimension of PCA-subspace, or the size of the windows taken from the images during classification.

In order to evaluate the results, the algorithm was tested on the training images and the detected positions were compared to the reference data. Depending on whether using four or five template sizes, 75% or 80% of the bomb craters are captured with the created template. After classification 70% to 64% of the bomb craters are detected, depending on configuration of the parameters. However, relevance is very low. At most about 31% of the windows classified as bomb craters truly are bomb craters.

When analyzing the false positives on test images, we found that certain structures in the images, like rooftops, shadows near streets, textures in fields or forest structures, often are wrongly classified as craters. In determining the craters not captured by the template, we could derive new classes of craters for further development of the algorithm. We also examined, whether or not it is possible to separate the test images by means of bomb crater detection into the categories weak, medium and strong bombing. 73% of the images were classed into the correct category. By increasing relevance and when annotating more images we expect better classification results.

Overall, this thesis presents a promising approach for bomb crater detection with many extension possibilities.

# Inhaltsverzeichnis

<b>Einleitung – Aufgabe und Gliederung</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Problembeschreibung . . . . .	2
1.2.1 Gestalt der Bombentrichter . . . . .	2
1.2.2 Anforderungen und Einschränkungen . . . . .	4
1.3 Lösungsstrategien . . . . .	5
1.3.1 Klassifikation ohne Vorauswahl . . . . .	5
1.3.2 Klassifikation mit Vorauswahl . . . . .	6
1.3.3 Bestehende Arbeiten . . . . .	7
1.4 Gliederung der Arbeit . . . . .	7
<b>Konzept – Lösungsverfahren und theoretische Grundlagen</b>	<b>9</b>
2.1 Gesamtkonzept . . . . .	9
2.1.1 Lernphase . . . . .	10
2.1.2 Arbeitsphase . . . . .	14
2.1.3 Erweiterung des Konzepts . . . . .	14
2.2 Theoretische Grundlagen . . . . .	15
2.2.1 Hauptkomponentenanalyse . . . . .	15
2.2.2 Fisher's lineare Diskriminanzanalyse . . . . .	17
<b>Umsetzung – Implementierung und Parameterwahl</b>	<b>19</b>
3.1 Bildmaterial und Vorarbeiten . . . . .	19
3.1.1 Vorliegendes Bildmaterial . . . . .	19
3.1.2 Verwendete Software . . . . .	20
3.1.3 Vorverarbeitung des Bildmaterials . . . . .	20
3.1.4 Annotation der Trainingsbilder . . . . .	21
3.1.5 Ausschneiden und Skalieren von Bildfenstern . . . . .	23
3.2 Kandidatensuche . . . . .	24
3.2.1 Erstellung des Templates . . . . .	24
3.2.2 Kandidatensuche mittels Kreuzkorrelation . . . . .	26
3.3 Erstellung der Trainingsmuster . . . . .	26
3.3.1 Auswahl der Trainingsmuster . . . . .	26
3.3.2 Anzahl und Verhältnis der Trainingsmuster . . . . .	27
3.4 Training . . . . .	28

3.4.1	Hauptkomponentenanalyse . . . . .	28
3.4.2	Fisher's lineare Diskriminanzanalyse . . . . .	29
3.4.3	Schätzung der Parameter der Verteilungen . . . . .	30
3.4.4	Klassifizierung . . . . .	31
3.5	Ausschnittsgröße bei der Klassifikation . . . . .	32
<b>Ergebnisse – Analyse und Bewertung des Algorithmus</b>		<b>34</b>
4.1	Allgemeines zur Bewertung . . . . .	34
4.2	Ergebnisse bei der Kandidatensuche . . . . .	36
4.2.1	Verschiedene Templateanzahlen . . . . .	36
4.2.2	Rotationsinvarianz des Templates . . . . .	38
4.3	Ergebnisse der Klassifikation . . . . .	39
4.4	Analyse der Fehlklassifikation . . . . .	41
4.4.1	False positives . . . . .	42
4.4.2	False negatives . . . . .	45
4.5	Tests mit neuen Bildern . . . . .	48
<b>Zusammenfassung und Ausblick – Was erreicht wurde und was noch fehlt</b>		<b>51</b>
5.1	Zusammenfassung . . . . .	51
5.2	Ausblick . . . . .	52
<b>Literaturverzeichnis</b>		<b>52</b>
<b>Abbildungsverzeichnis</b>		<b>54</b>
<b>Tabellenverzeichnis</b>		<b>56</b>
<b>Anhang</b>		<b>56</b>
<b>A Notation</b>		<b>57</b>
<b>B Bilder</b>		<b>59</b>
<b>C Hausarbeit zum Oberseminar</b>		<b>69</b>
<b>D Plakative Darstellung</b>		<b>80</b>

# 1. Einleitung

## Aufgabe und Gliederung

### 1.1 Motivation

Während des Zweiten Weltkriegs nahmen die alliierten Streitkräfte über 300000 Luftbilder von Nordrhein-Westfalen auf<sup>1</sup>. Sie dokumentierten damit ihre militärischen Erfolge und Misserfolge und nutzten sie, um das Kampfgebiet zu erkunden und ihre weitere Strategie zu planen [Drauschke et al., 2007]. Heute werden die Bilder unter anderem vom Kampfmittelbeseitigungsdienst der Bezirksregierung Arnsberg genutzt, um Blindgänger zu lokalisieren. Im Jahr 2006 wurden allein in Nordrhein-Westfalen knapp 1100 nicht detonierte Bomben entschärft<sup>2</sup>.

„Die Aufgaben des Kampfmittelbeseitigungsdienstes sind auch künftig aktuell und wichtig. So wurden im Jahr 2003 durch die bisherigen Kampfmittelräumdienste der drei Bezirksregierungen mehr als 2000 Baustellen untersucht und mehr als 2500 Gebiete anhand von rd. 150.000 Luftbildern ausgewertet. 28 Tonnen Explosivgewicht konnten allein in diesem Zeitraum unschädlich gemacht werden. Nach vorsichtigen Schätzungen wird man auch in den nächsten rd. 30 Jahren auf die Tätigkeit der Kampfmittelbeseitigungsdienste nicht verzichten können.“<sup>3</sup>

Die Blindgängerrate bei einem Luftangriff der Alliierten im Zweiten Weltkrieg lag laut Lücke et al. [2007] bei zehn bis fünfzehn Prozent der Abwurfmenge. Aufgrund der großen Menge an Bildern und der zeitaufwendigen Analyse ist es wünschenswert, die Suche nach Blindgängern in den Luftbildern zu automatisieren. Dies ist allerdings eine sehr komplexe Aufgabe, da die von Blindgängern erzeugten Einschlaglöcher meist sehr klein und unauffällig sind. Es gibt zwar ein Verfahren von Lücke et al. [2007], das Blindgänger halbautomatisch mit Hilfe von neuronalen Netzen detektiert, es ist aber wegen der hohen Fehlklassifikationsrate für den Kampfmittelbeseitigungsdienst der Bezirksregierung Arnsberg nicht verwendbar<sup>4</sup>.

---

<sup>1</sup>siehe Homepage des Innenministeriums NRW, <http://www.im.nrw.de/sch/725.htm>, zuletzt aufgerufen am 14.04.2008

<sup>2</sup>siehe Homepage des Innenministeriums NRW, <http://www.im.nrw.de/sch/725.htm>, zuletzt aufgerufen am 14.04.2008

<sup>3</sup>siehe Homepage der Bezirksregierung Arnsberg, [http://www.bezreg-arnsberg.nrw.de/aktuell/presseArchiv/2004/april/051\\_04.html](http://www.bezreg-arnsberg.nrw.de/aktuell/presseArchiv/2004/april/051_04.html), zuletzt aufgerufen am 13.06.2008

<sup>4</sup>laut Dr. Ansgar Brunn in einer Besprechung am 14.01.2008

Die manuelle Inspektion der Luftbilder nach Blindgängern würde erheblich erleichtert werden durch ein Verfahren, das die Bombentrichter automatisch lokalisiert. Da sich die Blindgänger nur in Gebieten befinden können, die bombardiert wurden, kann man aus dem Vorhandensein von Trichtern, die von detonierten Bomben erzeugt wurden, darauf schließen, dass in dem Gebiet auch Blindgänger liegen. Nicht bombardierte Gebiete könnten so bereits ohne menschliche Betrachtung aus der Suche nach Blindgängern ausgeschlossen werden. Aufgabe der Bachelorarbeit ist es, ein Verfahren zu entwickeln und umzusetzen, das die automatische Detektion von Bombentrichtern im ländlichen Raum realisiert.

## 1.2 Problembeschreibung

In diesem Abschnitt wird die Problemstellung der Bachelorarbeit näher untersucht, und Anforderungen an ein Programm zur Bombentrichterdetektion werden spezifiziert.

Für die Bearbeitung der Bachelorarbeit stehen digitale Luftbilder von Nordrhein-Westfalen zur Verfügung. Sie wurden von den Alliierten überwiegend in den Jahren 1939 bis 1945 zu unterschiedlichen Jahres- und Tageszeiten aufgenommen. Sie unterscheiden sich in ihrer Qualität, ihrem Maßstab, Aufnahmesituation, Kontrast, Helligkeit und Größe. Es liegt zwar eine Sortierung der Bilder nach Landkreisen vor, allerdings fehlt die genaue Georeferenzierung. Bekannt ist aber Datum und Uhrzeit des jeweiligen Bildfluges, zu dem sie gehören, sowie der ungefähre Maßstab. Die Anzahl an Bombentrichtern auf einem Bild reicht von Null bis hin zu mehreren Tausend. Einzelne Bombentrichter sind selten, meist schlugen bei einer Bombardierung viele Bomben in der näheren Umgebung ein.

Ziel der Bombentrichterdetektion ist die Erstellung einer „Bombentrichter-Dichte-Karte“, die darstellt, welche Gebiete wie stark bombardiert wurden. Damit kann dem Kampfmittelbeseitigungsdienst die Suche nach Blindgängern erleichtert werden. Um diese Karte zu erstellen, sind möglichst die Positionen der Mittelpunkte der auf Bildern erkennbaren Bombentrichter im Bildkoordinatensystem zu finden.

### 1.2.1 Gestalt der Bombentrichter

Um einen Ansatz für die Detektion zu finden, ist zunächst zu klären, in welcher Gestalt die Bombentrichter in den Bildern auftreten. Im Wesentlichen sind sie im Bild kreisrunde Objekte. Der Erdauswurf, der beim Einschlag und der Detonation entstand, zeigt sich oft in einem charakteristischen ausgefransten, inhomogenen Kranz rund um das Trichterinnere. Die Bombentrichter unterscheiden sich jedoch in ihrem Aussehen sowohl innerhalb eines Bildes als auch zwischen verschiedenen Bildern. Die Ursachen für die Erscheinungsvariationen können wir in drei Kategorien einteilen: physikalische Ursachen, Aufnahmesituation und Qualitätsunterschiede der Bilder.

Eine wichtige physikalische Ursache für die unterschiedliche Gestalt ist die Bodenbeschaffenheit, die je nach Gebiet variiert. Zum Beispiel erzeugen Bomben, die in Lehmboden einschlagen, andere Trichter, als solche, die in Sandboden einschlagen. Eine zweite Ursache ist die unterschiedliche Landschaftsgestaltung. Bombentrichter haben im Wald eine andere Form als auf Äckern oder bebauten Flächen. Zusätzlich gibt es Variationen im Aussehen der Bombentrichter in der Natur durch die unterschiedlichen Gewichtsklassen der Bomben. Die schwersten abgeworfenen

Bomben wogen etwa 6 Tonnen, die leichtesten etwa 2 Kilogramm<sup>5</sup>.

Ein großer Einflussfaktor auf das Aussehen der Trichter im Bild ist die Aufnahmesituation. Aufgrund der unterschiedlichen Flugzeiten variiert der Sonnenstand zum Zeitpunkt der Aufnahme. Der meist näherungsweise elliptische oder sichelförmige Schattenwurf, der für viele Trichter charakteristisch ist, ist deshalb von Bild zu Bild anders ausgerichtet. Es ist jede Ausrichtung der Schattenellipsen und -sicheln im Bild möglich, da auch die Flugrichtung variiert. Manche Trichter waren zum Zeitpunkt der Aufnahme mit Wasser gefüllt. Sie haben dann keinen (oder einen sehr schmalen) sichelförmigen Schatten im Inneren, sondern reflektieren das Sonnenlicht und sehen anders aus als nicht gefüllte Trichter. Unmittelbaren Einfluss auf die Größe des Trichters im Bild hat die Flughöhe, und damit der Bildmaßstab. Nicht berücksichtigt werden muss die Neigung der Kamera in Bezug auf die Landschaft. Zwar befinden sich unter den Luftbildern der Alliierten auch Schrägaufnahmen, auf denen Bombentrichter zu erkennen sind, im Rahmen der Bachelorarbeit wurde die Suche nach Bombentrichtern aber auf nahezu senkrechte Aufnahmen beschränkt.

Die dritte Ursachenkategorie sind Qualitätsunterschiede im Bild, zu denen auch Helligkeit und Kontrast zählen. Globale Helligkeitsunterschiede, die aus der Aufnahmesituation resultieren, wurden allerdings vorher mit Hilfe des Dodging-Verfahrens nach Drauschke et al. [2007], das auf abschnittsweiser Histogrammverebnung beruht, weitgehend angepasst (siehe auch Kapitel 3.1.3). Damit besteht eine globale Vergleichbarkeit der Bilder bezüglich ihrer Grauwertverteilung.

Die beschriebenen Ursachen resultieren in Variationen der abgebildeten Bombentrichter in Größe, Form, Helligkeit und Kontrast, wie in Abbildung 1.1 und 1.2 beispielhaft dargestellt. Das Aussehen der nahen Umgebung der Einschlaglöcher variiert aufgrund der unterschiedlichen Ausprägung des Erdauswurfs, der bei der Detonation entstand, ebenfalls. Des Weiteren ähneln

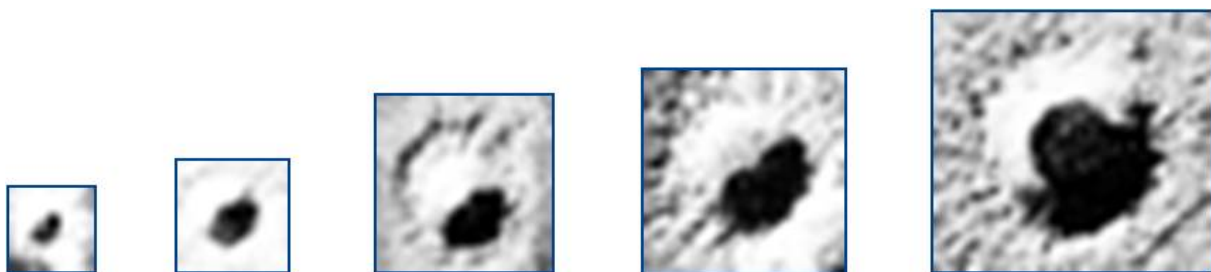


Abbildung 1.1: Ausschnitte der gleichen Auflösung aus einem Luftbild zeigen die Größenvariationen der Trichter

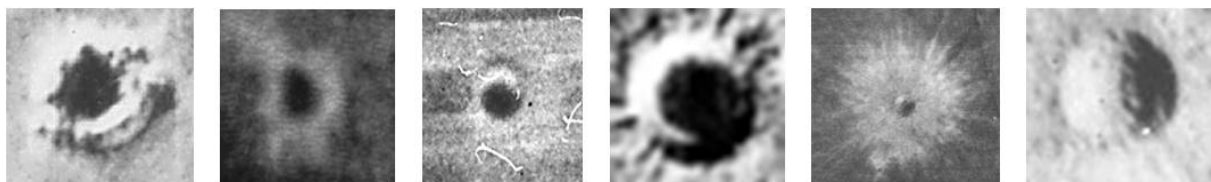


Abbildung 1.2: Ausschnitte aus verschiedenen Bildern zeigen die Formvariationen der Trichter (nach dem Dodging)

<sup>5</sup>siehe „Arbeitshilfen Kampfmittelräumung“ des Bundesministeriums für Verkehr, Bau und Stadtentwicklung und des Bundesministeriums für Verteidigung, <http://www.arbeitshilfen-kampfmittelraeumung.de/Kapitel/1197022191818.html>, zuletzt aufgerufen am 22.06.2008

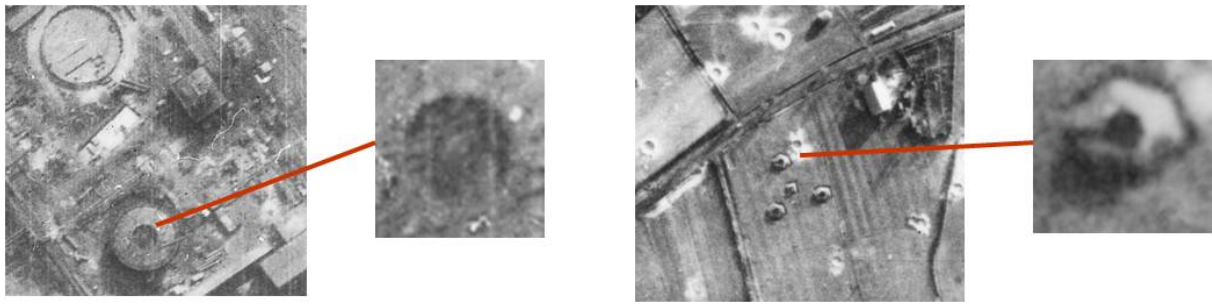


Abbildung 1.3: Trichterähnliche Objekte

die Trichter anderen Objekten im Bild (Abbildung 1.3). Vor allem in Gebieten, die stark bombardiert wurden, gibt es Bombentrichter, die von anderen überlappt werden, so dass die kreisrunde Form nicht immer gegeben ist, und teilweise nur Fragmente von Trichtern sichtbar sind.

Bei der Betrachtung einiger Luftbilder stellte sich heraus, dass die Form des Trichters unabhängig von seiner Größe im Bild ist (siehe auch Abbildung 1.1: Die Form ist trotz verschiedener Größe ähnlich). Offenbar spielt die Beschaffenheit des Untergrunds sowie die Aufnahmesituation eine größere Rolle für die Gestalt des Trichters als das Gewicht der Bombe. Für die Betrachtung der Form unabhängig von der Größe spricht auch, dass der Maßstab der Bilder nicht gleich ist; Bomben einer Gewichtsklasse haben daher in verschiedenen Bildern nicht notwendigerweise die gleiche Trichtergröße.

### 1.2.2 Anforderungen und Einschränkungen

Die obige Erläuterung zeigt, dass es für die Bombentrichterdetektion erforderlich ist, einen Algorithmus zu implementieren, der mit der großen Bandbreite in der Gestalt (Form und Größe) der Bombentrichter umgehen kann. Er sollte möglichst erweiterbar sein, falls neues Bildmaterial und damit weitere Trichterausprägungen hinzukommen. Außerdem ist es notwendig, Trichter von anderen, ähnlichen Objekten abgrenzen zu können. Aufgrund der Komplexität, die sich bei der Hinzunahme der Detektion von sich überlappenden Trichtern ergäbe, wurde die Aufgabe der Bachelorarbeit auf die Detektion von einzelnen Trichtern beschränkt.

Die Bombentrichterdetektion soll später den Kampfmittelbeseitigungsdienst bei der Suche nach Blindgängern unterstützen. Blindgänger können in jedem bombardierten Gebiet vorhanden sein, ihre Zahl ist nicht von der Art der Bomben oder des Bodens, in den sie einschlagen, abhängig. Es ist deshalb für diese Arbeit nicht notwendig, die Bombentrichter nach ihrer Art oder Größe zu unterscheiden. Bei der Detektion sollen sie lediglich lokalisiert werden.

Es ist nicht zu erwarten, dass ein Algorithmus, der auf Einzelbildern arbeitet, optimale Ergebnisse liefert, d.h. jeden Trichter und nur Trichter detektiert. Da das Bildmaterial nicht optimal ist, ist es selbst für einen geübten Betrachter oft nicht ohne Weiteres möglich, zu entscheiden, ob es sich bei einem Objekt um einen Trichter, einen Blindgänger, oder ein anderes Objekt handelt. Beim Kampfmittelbeseitigungsdienst der Bezirksregierung Arnsberg werden deshalb Zeitreihenanalysen durchgeführt. Die Experten betrachten Luftbilder des gleichen Gebiets, die zu verschiedenen Zeiten (möglichst vor und nach der Bombardierung) aufgenommen wurden. Ein Objekt, das nur auf den Bildern nach der Bombardierung zu sehen ist, ist mit hoher Wahrscheinlichkeit ein Bombentrichter. Die Zeitreihenanalyse ist allerdings nicht automatisiert möglich, da



die Luftbilder nicht orientiert sind.

Die Stereobildauswertung ist ein weiteres Mittel, dessen sich menschliche Betrachter bedienen können, um Bombentrichter besser von anderen Objekten abgrenzen zu können. Sie können so beispielsweise Objekte ausschließen, die sich bis zu einer gewissen Höhe über dem Erboden ausdehnen (z.B. Bäume). Da die Bilder nicht orientiert sind, ist es nicht möglich, automatisch ein 3D-Modell zu erstellen, und die Bombentrichtersuche dort durchzuführen. Damit bleibt die Stereobildauswertung zur Trichterdetektion ein Mittel, das nur von menschlichen Betrachtern durchgeführt werden kann.

## 1.3 Lösungsstrategien

Es gibt verschiedene Ansätze, die automatische Bombentrichterdetektion zu realisieren. In diesem Abschnitt werden einige dieser Ansätze auf Verwendbarkeit für die Lösung des oben beschriebenen Problems untersucht und der umgesetzte Ansatz begründet.

Um in einem Bild die Bombentrichter zu detektieren ist es allgemein notwendig, Bildausschnitte zu klassifizieren. Da die Trichter nicht nach ihrer Art unterschieden werden sollen, werden die Bildausschnitte in zwei Klassen eingeordnet: entweder in die Klasse der Trichter oder in die Klasse der „Nicht-Trichter“ (im Folgenden: Hintergrund). Die Klassifikation kann entweder ohne Vorauswahl der Bildausschnitte oder mit Vorauswahl durchgeführt werden. Im ersten Fall wird an jeder Pixelposition des Bildes ein Ausschnitt (oder mehrere Ausschnitte unterschiedlicher Größe) genommen und klassifiziert. Im zweiten Fall werden bei Anwendung des Algorithmus auf ein Bild zunächst in einem ersten Schritt bestimmte Bildfenster ausgewählt, und in einem zweiten Schritt nur diese klassifiziert. Optimalerweise sollten in der Vorauswahl alle Trichter des Bildes enthalten sein, damit nicht bereits durch die Vorauswahl Bombentrichter von der Detektion ausgeschlossen werden.

### 1.3.1 Klassifikation ohne Vorauswahl

Der Vorteil bei der Klassifikation aller möglichen Bildausschnitte liegt darin, dass kein Trichter bereits durch die Vorauswahl ausgeschlossen wird. Jeder Ausschnitt wird als potentieller Trichter betrachtet und nur aufgrund der Einordnung in die Klasse Hintergrund nicht als solcher klassifiziert.

Bei einer Untersuchung dieses Ansatzes, bei dem ein Ausschnitt an jeder Bildposition klassifiziert wurde, ergab sich allerdings aufgrund der großen Anzahl an Ausschnitten ein Laufzeit- bzw. Speicherplatzproblem. Somit ist die Klassifikation von sämtlichen Bildausschnitten im Rahmen der Bachelorarbeit nicht durchführbar.

Selbst wenn die Laufzeit- und Speicherplatzprobleme gelöst würden, wäre der Ansatz im Rahmen dieser Arbeit nicht realisierbar, denn es zeigte sich bei der Untersuchung auch, dass die beiden Klassen keine einfachen Cluster bilden. Somit wäre die Anwendung eines komplexen Klassifikationsverfahrens nötig, um die Klassen Trichter und Hintergrund zu trennen, das in der Bachelorarbeit aus Zeitgründen nicht erarbeitet werden kann. Deshalb wurde das zweistufige Verfahren mit einer Vorauswahl der Bildausschnitte, die kompakte Cluster bilden, und einem linearen Klassifikationsverfahren gewählt. Das Klassifikationsverfahren kann bei Bedarf in

späteren Arbeiten durch ein leistungsfähigeres und komplexeres ersetzt werden, um die Klassifikationsergebnisse zu verbessern.

### 1.3.2 Klassifikation mit Vorauswahl

Es existiert eine Vielzahl an möglichen Verfahren für die Vorauswahl der Bildfenster, die in die Klassifikation eingehen. Einige wollen wir kurz im Hinblick auf die Eignung für die Trichterdetektion betrachten:

- Wenn wir davon ausgehen, dass die Bombentrichter im Wesentlichen kreisförmige Gestalt haben und sich deutlich von der Umgebung abheben, so können wir zur Detektion einen *Punkt-Operator* (z.B. Punkt-Operator nach Förstner [1994]) verwenden. Somit könnten die Trichter mit dem Punkt-Operator bei entsprechender Fenstergröße als markante Punkte extrahiert werden. Die unterschiedliche Größe der Trichter könnte durch die mehrfache Anwendung des Operators unter Verwendung von verschiedenen Fenstergrößen berücksichtigt werden. Ein Nachteil bei der Verwendung eines Punkt-Operators zur Vorauswahl der Bildfenster ist die fehlende Möglichkeit zur Erweiterung. Er kann nicht einfach modifiziert werden, so dass Trichter, die zunächst nicht als markante Punkte erkannt wurden, nach der Erweiterung doch extrahiert werden.
- Der *SIFT-Operator* von Lowe [2004] ist ein Verfahren, das markante Punkte mit einer Skala sowie ihrer Orientierung und einer Beschreibung ihrer näheren Umgebung extrahiert. Beschränkte man die Auswahl der Bildregionen dann auf die vom SIFT-Operator gelieferten Punkte mit den für Bombentrichter relevanten Skalen, wäre auch dieses Verfahren zur Vorauswahl von Bildausschnitten geeignet. Auch hier ist, wie bei den Punkt-Operatoren, die Erweiterung nicht ohne Weiteres möglich.
- Eine weitere Möglichkeit zur Vorauswahl von Bildausschnitten ist die *Extraktion von homogenen Regionen*, z.B. mit dem Wasserscheiden-Algorithmus nach Vincent and Soille [1991], oder der Blob-Detektion nach Lindeberg [1994]. Nach der Segmentierung des Bildes könnte zum Beispiel eine Suche nach kreisförmigen Flächen mit einer Größe im Bereich der Trichtergröße durchgeführt werden, und diese als Bildausschnitte klassifiziert werden. Das Problem bei vielen Verfahren zur Extraktion von Regionen ist die Übersegmentierung, d.h. das Bild wird aufgrund von Beleuchtungsschwankungen und Schattenwurf in zu viele Regionen zerlegt. Gerade bei Trichtern mit sichelförmigen Schattenwurf wäre die Zerlegung in zwei oder mehrere Regionen wahrscheinlich. Zudem zeigt die obige Darstellung der verschiedenen Trichtergestalten, dass die Trichter auch ohne sichelförmigen Schattenwurf meist keine homogenen Regionen sind. Viele sind von einem Erdauswurf umgeben, der sich im Bild als inhomogene Fläche darstellt. Oft ist es aber gerade dieser deutlich sichtbare Erdauswurf, der es ermöglicht, den Trichter als solchen zu identifizieren. Somit wäre die Extraktion homogener Regionen voraussichtlich kein geeignetes Verfahren zur Vorauswahl der Bildausschnitte.
- Die *Kreuzkorrelation* des Bildes mit einem geeigneten Template liefert eine Korrelationsmatrix, die die Ähnlichkeit des Templates mit dem Bildausschnitt an der entsprechenden

Position repräsentiert. Wird als Template ein Bild verwendet, das einen möglichst allgemeinen und repräsentativen Trichter zeigt, könnten Positionen der Maxima dieser Korrelationsmatrix Trichterpositionen im Bild sein. Die Bildfenster um diese Positionen könnten also klassifiziert werden. Die Größenvariationen der Trichter könnten durch Verwendung von mehreren Templates in verschiedenen Größen berücksichtigt werden. Ein großer Vorteil dieses Verfahrens ist die einfache Erweiterungsmöglichkeit. Das Template kann modifiziert werden, um mehr Bildausschnitte zu bekommen, oder es können mehrere Templates verwendet werden, die die verschiedenen Trichterausprägungen repräsentieren. Damit kann die Suche nach Bildfenstern so optimiert werden, dass möglichst viele Trichter eines Bildes in der Vorauswahl enthalten sind.

Aufgrund der guten Erweiterungsmöglichkeit wird in dieser Arbeit die kreuzkorrelationsbasierte Vorauswahl von Bildausschnitten und anschließende Klassifikation realisiert.

### 1.3.3 Bestehende Arbeiten

Im zivilen Bereich gibt es bisher keine Vorarbeiten zur automatischen Detektion von Bombentrichtern in Luftbildern. Dass die Suche nach Kandidaten mittels Kreuzkorrelation ein vielversprechender Ansatz ist, zeigen Wenzel and Förstner [2008a]. In ihrem Verfahren wird in einem Bild zunächst ein Bildausschnitt als Template per Hand ausgewählt, und dann die Kreuzkorrelation rekursiv durchgeführt. Alle mit dem Template gefundenen Bildfenster werden also als weitere Templates für neue Kreuzkorrelationen verwendet. Bei Wenzel and Förstner [2008a] wird dieses Verfahren zwar in einem anderen Kontext genutzt, dennoch belegt es, dass die Kreuzkorrelation mit verschiedenen Templates gut für die Detektion bestimmter Objekte, z.B. Trichter, geeignet ist. Abbildung 1.4 zeigt links den Bildausschnitt, auf den das Verfahren angewendet wurde, rechts sind blau umrandet die Bombentrichter dargestellt, die mit dem Ausgangstemplate (rot umrandet) bei der Rekursionstiefe 2 (und nach einem Clustering) gefunden wurden.

## 1.4 Gliederung der Arbeit

Nachdem in der Einleitung das Problem beschrieben und der gewählte Ansatz zur Trichterdetektion begründet wurde, folgt im Kapitel 2 die Beschreibung des umzusetzenden Konzepts und die theoretischen Grundlagen, die dem Konzept zugrunde liegen. Die dort eingeführte Notation ist zur Übersicht noch einmal tabellarisch im Anhang zusammengefasst. Die Umsetzung wird in Kapitel 3 detailliert erläutert und die Ergebnisse der Arbeit werden in Kapitel 4 vorgestellt. Kapitel 5 schließt die Arbeit mit einer Zusammenfassung und einem Ausblick ab.

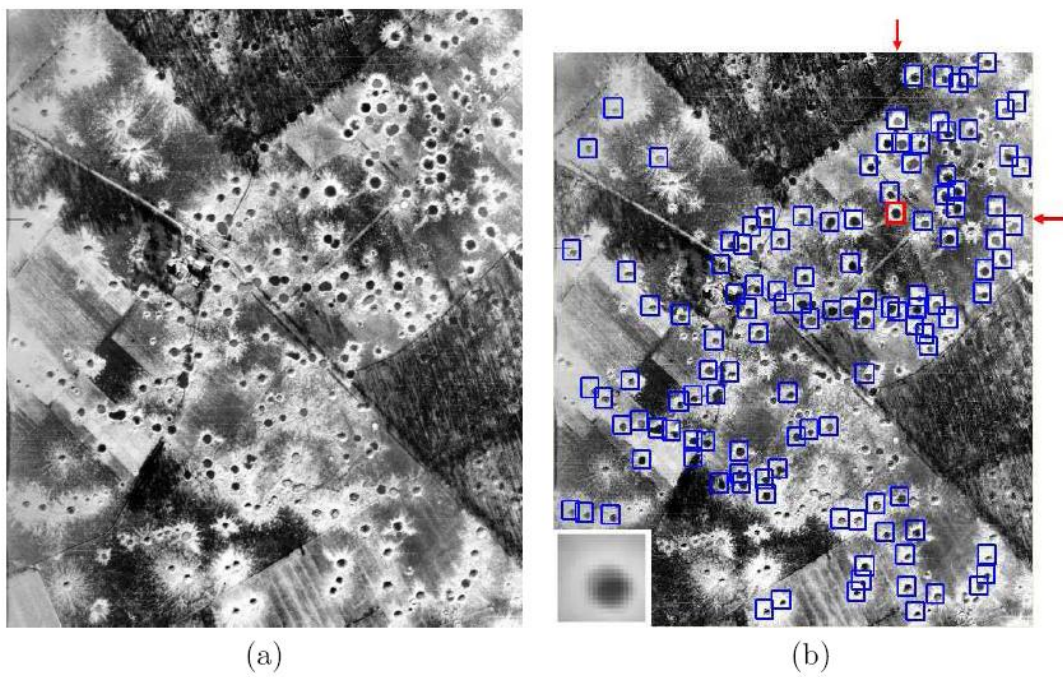


Abbildung 1.4: Ergebnis der Suche nach Trichtern mittels Kreuzkorrelation. Aus Wenzel and Förstner [2008b].

# 2. Konzept

## Lösungsverfahren und theoretische Grundlagen

In diesem Kapitel konkretisieren wir den in der Einleitung vorgestellten Ansatz zur Detektion der Bombentrichter in einem digitalen Grauwertbild. Im Anschluss an die detaillierte Vorstellung des Gesamtkonzepts werden die theoretischen Grundlagen erläutert, die dem Konzept zu Grunde liegen.

### 2.1 Gesamtkonzept

Gegeben sind digitale Luftbilder und gesucht sind die Positionen von möglichen Bombentrichtern. Wie im Kapitel 1.2.1 beschrieben, unterscheiden sich die Bombentrichter erstens in ihrer Größe und zweitens in ihrer Helligkeit und Ausprägung. Deshalb genügt es nicht, die Trichter allein durch Kreuzkorrelation des Bildes mit einem Template, das einen Trichter zeigt, zu suchen.

In Kapitel 1.3 wurde bereits dargelegt, dass in dieser Arbeit die Detektion durch eine Vorauswahl von Bildausschnitten und anschließende Klassifikation realisiert wird. Die Vorauswahl geschieht durch Kreuzkorrelation des Bildes mit einem allgemeinen, repräsentativen Bombentrichter-Template in verschiedenen Größen. Die Bildfenster um die so gefundenen möglichen Trichterpositionen herum werden als Trichter oder Hintergrund klassifiziert. Aus der Vielzahl an möglichen Klassifikationsverfahren wurde in dieser Arbeit die Klassifikation durch Bestimmung der Wahrscheinlichkeitsdichten der Klassen Trichter und Hintergrund ausgewählt. Dafür nehmen wir an, dass sich die verschiedenen Ausprägungsformen der kleinen Trichter – abgesehen von der Skalierung – nicht von denen der großen Trichter unterscheiden. Weiterhin gehen wir davon aus, dass sich die Gestaltvariationen der Trichter in der statistischen Verteilung einer repräsentativen Auswahl von Bildausschnitten mit Bombentrichtern ausdrücken. Bei der Klassifikation verwenden wir als Trainingsmuster eine Menge von quadratischen Bildausschnitten mit Bombentrichtern und Hintergrund. Die betrachteten Merkmale sind die Intensitätswerte der Bildausschnitte. Aus der Schätzung der Parameter der Verteilung der Klassen Trichter und Hintergrund können die Wahrscheinlichkeitsdichtefunktionen, die für die Klassifizierung verwendet werden, berechnet werden.

Das Gesamtkonzept der Bombentrichterdetektion, das in der Bachelorarbeit umgesetzt wird, gliedert sich in zwei Phasen: die Lernphase und die Arbeitsphase. Die Lernphase enthält das

Erstellen des Templates für die Kreuzkorrelation und das Lernen der Verteilung der Merkmale aus den Trainingsmustern. In der Arbeitsphase führen wir die Kandidatensuche mit dem in der Lernphase erstellten Template durch und klassifizieren die gefundenen Kandidaten. Im Folgenden werden die beiden Teile des Gesamtkonzepts genauer erläutert.

### 2.1.1 Lernphase

Aus einer Trainingsmenge von Luftbildern werden  $T$  Bildfenster  $\mathbf{T}'_t$ ,  $t = 1, \dots, T$ , mit Trichtern ausgeschnitten. Zur Auswahl der Fenster werden in allen Trainingsbildern Bombentrichter manuell markiert. Die Ausschnittsgröße richtet sich nach der Größe des Trichters.

#### 2.1.1.1 Erstellung der Templates

Aus den  $\mathbf{T}'_t$  soll ein allgemeines, repräsentatives Template erstellt werden. Es ist naheliegend, den Mittelwert, also den mittleren Bombentrichter, als Template zur Kandidatensuche zu verwenden. Dafür werden die Trichterfenster auf eine Größe skaliert, und der Mittelwert über alle skalierten Trichterfenster gebildet. Die Bildmatrizen der auf die Größe  $I \times J$  skalierten Trichterausschnitte bezeichnen wir mit  $\mathbf{T}_t$ . Der Mittelwert aller  $\mathbf{T}_t$  ergibt das Muster  $\mathbf{M}$ :

$$\mathbf{M} = \frac{1}{T} \sum_{t=1}^T \mathbf{T}_t$$

Für die Kreuzkorrelation wird  $\mathbf{M}$  auf  $S$  Größen skaliert. Damit erhalten wir die Templates  $\mathbf{M}_1, \dots, \mathbf{M}_S$ . Abbildung 2.1 veranschaulicht den Ablauf.

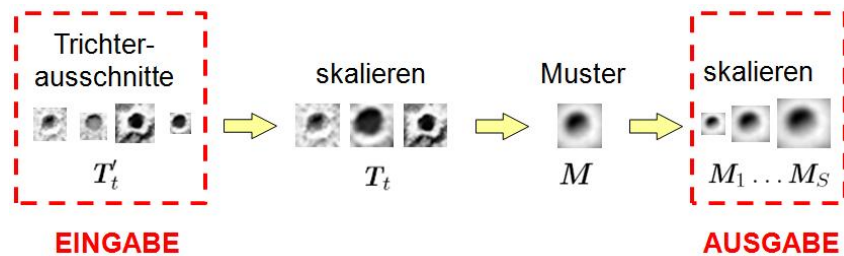


Abbildung 2.1: Erstellung der Templates für die Kandidatensuche mittels Kreuzkorrelation

#### 2.1.1.2 Auswahl der Trainingsmuster

Zur Bestimmung der Verteilungsparameter für die Klassen Trichter und Hintergrund werden repräsentative Trainingsmuster benötigt. Als zu klassifizierende Kandidaten erhalten wir später bei der Kreuzkorrelation mit den Templates  $\mathbf{M}_1, \dots, \mathbf{M}_S$  Bildausschnitte, die eine Ähnlichkeit mit dem Template haben und deshalb trichterähnlich aussehen. Unter diesen Kandidaten werden sich allerdings sowohl Bildausschnitte mit Trichtern als auch mit Hintergrund befinden. Um diese zu trennen, wird die Klassifikation durchgeführt.

Um möglichst repräsentative Trainingsmuster zu erhalten, wird die Kandidatensuche mit Trainingsbildern durchgeführt, bei denen die Positionen der Trichter bekannt sind. Durch Vergleich der gefundenen Kandidaten mit den wahren Trichterpositionen können diejenigen Kandidaten

ermittelt werden, die Trichter darstellen. Diese verwenden wir als Trainingsmuster für die Klasse Trichter. Beim Training des Klassifikators für die Klasse Hintergrund müssen Trainingsmuster verwendet werden, die zwar Trichtern ähneln, jedoch keine sind. Diese erhalten wir als diejenigen Kandidaten, die mit dem Template auf den Trainingsbildern gefunden wurden, jedoch auf diesen nicht als Trichter markiert sind.

Auf einem Grauwertbild  $\mathbf{B} \in \mathbb{R}^{U \times V}$  der Trainingsmenge werden also  $S$  Kreuzkorrelationen mit den  $S$  Templates  $\mathbf{M}_s$  durchgeführt und die Maxima der Korrelationsfunktion

$$\rho(\mathbf{B}, \mathbf{M}_s; u, v) = \frac{\sigma(\mathbf{B}, \mathbf{M}_s; u, v)}{\sigma(\mathbf{B}; u, v) \cdot \sigma(\mathbf{M}_s)}$$

bestimmt. Die Maximumsuche liefert für jede der  $S$  Templategrößen mögliche Positionen für Bombentrichter im Bild. Aus dem Eingabebild  $\mathbf{B}$  werden an diesen Positionen insgesamt  $P$  Bildfenster  $\mathbf{G}'_p$  in bis zu  $S$  Größen ausgeschnitten. Durch Vergleich mit den wahren Bombentrichterpositionen werden die  $\mathbf{G}'_p$  in zwei Gruppen aufgeteilt: diejenigen Bildfenster, die Trichter zeigen, und diejenigen, die Hintergrund zeigen. Die  $M$  so ermittelten Trichterausschnitte bezeichnen wir mit  $\mathbf{T}'_m$ , und die  $L$  Hintergrundausschnitte mit  $\mathbf{H}'_l$ . Um mit den zunächst verschiedenen großen Ausschnitten  $\mathbf{T}'_m$  und  $\mathbf{H}'_l$  arbeiten zu können, skalieren wir diese auf die Größe  $I \times J$  und erhalten die Trainingsmuster  $\mathbf{T}_m$  und  $\mathbf{H}_l$  für die Klassen Trichter und Hintergrund.

### 2.1.1.3 Bestimmung der Verteilungsparameter

Die Idee bei der Klassifikation liegt darin, aus den Trainingsmustern  $\mathbf{T}_m$  und  $\mathbf{H}_l$  die Parameter der statistischen Verteilungen der Klassen Trichter und Hintergrund zu schätzen und die mit dem Template gefundenen Kandidaten mit Hilfe dieser zu klassifizieren.

Die Merkmale zur Bestimmung der Klassenzugehörigkeit sind die Intensitätswerte der Bildausschnitte. Jeder der insgesamt  $N = M + L$  Bildausschnitte  $\mathbf{T}_m$  und  $\mathbf{H}_l$  besitzt damit  $D = I \times J$  Merkmale. Wir betrachten die  $\mathbf{T}_m$  und  $\mathbf{H}_l$  also als Merkmalsvektoren  $\mathbf{t}_m = \text{vec}(\mathbf{T}_m)$  und  $\mathbf{h}_l = \text{vec}(\mathbf{H}_l)$  im  $D$ -dimensionalen Raum. Der  $\text{vec}$ -Operator ist dabei der Operator, der die Bildmatrix in einen Vektor umwandelt, indem er die Spalten der Bildmatrix untereinander anordnet.

Beispiel:

$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \end{bmatrix} \quad \text{vec}(\mathbf{X}) = \begin{bmatrix} x_{11} \\ x_{21} \\ x_{12} \\ x_{22} \\ x_{13} \\ x_{23} \end{bmatrix} = \mathbf{x}$$

Aus der Matrix  $\mathbf{X}$  mit den Elementen  $x_{11}, x_{12}, x_{13}, x_{21}, x_{22}$  und  $x_{23}$ , wird mit dem  $\text{vec}$ -Operator der Spaltenvektor  $\mathbf{x}$ .

Die Dimension  $D$  des Merkmalsraums wird durch die quadratische Abhängigkeit von der Ausschnittsgröße schnell sehr groß, bei einem Ausschnitt der Größe  $10 \times 10$  Pixel ist  $D = 100$ , bei

$50 \times 50$  Pixeln bereits  $D = 2500$ . Aufgrund der begrenzten Anzahl an manuell ausgewählten Trainingsmustern<sup>1</sup> ist der hochdimensionale Merkmalsraum aber extrem dünn besetzt. Es ist daher nicht möglich, die Parameter der beiden Verteilungen direkt im  $D$ -dimensionalen Raum zu bestimmen.

Das Problem kann dadurch gelöst werden, dass die Trainingsmuster in einen  $R$ -dimensionalen Raum ( $R < D$ ) projiziert werden, der dann dichter besetzt ist und in dem die beiden Klassen möglichst gut trennbar sind. Ein Werkzeug, mit dem die Projektionsparameter bestimmt werden können, die für eine gute Trennbarkeit der Klassen im Unterraum sorgen, ist die *lineare Diskriminanzanalyse nach Fisher*<sup>2</sup>. Die Fisher'sche Diskriminanzanalyse benötigt allerdings für jede Klasse eine reguläre Kovarianzmatrix der eingehenden Daten. Ist die Anzahl  $M$  der Trainingsmuster mit Trichtern kleiner als die Dimension  $D$  des Merkmalsraums, so kann der Rang der  $D \times D$ -Kovarianzmatrix der  $\mathbf{t}_m$  nur  $M$  (und nicht voll) sein. Die Voraussetzung für die Durchführung einer Fisher'schen Diskriminanzanalyse wäre damit nicht erfüllt. In diesem Fall ist  $M$  zwar geringfügig größer als  $D$ , trotzdem ist eine numerisch sichere Bestimmung der Parameter der LDA nicht gewährleistet. Daraus folgt, dass wir die Dimension des Merkmalsraums bereits vorher reduzieren müssen. Die *Hauptkomponentenanalyse*<sup>3</sup> projiziert die gegebenen  $D$ -dimensionalen Daten so in einen  $K$ -dimensionalen Unterraum, dass möglichst wenig Information verloren geht. Sie ist damit für die Reduktion der Dimension der Trainingsdaten gut geeignet.

Die Trainingsmuster projizieren wir also mit Hilfe der PCA zunächst in einen  $K$ -dimensionalen Raum. Mit den projizierten  $K$ -dimensionalen Trainingsmustern wird danach eine LDA durchgeführt, mit deren Ergebnissen diese in den  $R$ -dimensionalen Raum projiziert werden, wo die Klassen Trichter und Hintergrund gut trennbar sind. Abbildung 2.2 veranschaulicht den Ablauf, der im Folgenden mathematisch dargestellt wird.

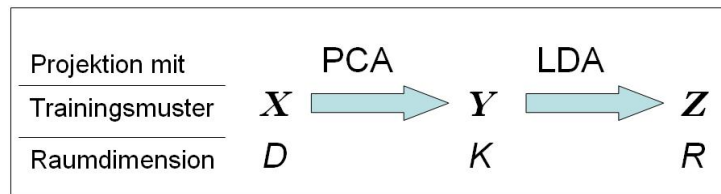


Abbildung 2.2: Ablauf der Dimensionsreduktion des Merkmalsraums für die Trainingsmuster

Die  $N$  Merkmalsvektoren  $\mathbf{t}_m$  und  $\mathbf{h}_l$  fassen wir zu einer Matrix  $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_N] \in \mathbb{R}^{D \times N}$  zusammen. Die Hauptkomponentenanalyse liefert eine Matrix  $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_D] \in \mathbb{R}^{D \times D}$  mit den  $D$  Eigenvektoren der Kovarianzmatrix der Trainingsdaten  $\mathbf{X}$ , sowie die zu den  $\mathbf{u}_d$  gehörenden Eigenwerte  $\lambda_d$ ,  $d = 1, \dots, D$ . Die ersten  $K$  Eigenvektoren nutzen wir nun dazu, die Trainingsmuster  $\mathbf{x}_n$  in den  $K$ -dimensionalen Unterraum zu projizieren. Die Projektion erfolgt mit der Formel:

$$\mathbf{y}_n = \mathbf{U}_K^T \cdot (\mathbf{x}_n - \boldsymbol{\mu}_x) \quad \boldsymbol{\mu}_x = \frac{1}{N} \sum_{n=1}^N \mathbf{x}_n$$

wobei  $\mathbf{U}_K = [\mathbf{u}_1, \dots, \mathbf{u}_K] \in \mathbb{R}^{D \times K}$  die Matrix mit den ersten  $K$  Eigenvektoren der Hauptkompo-

<sup>1</sup>Es wurden für die Berechnung etwa 900 Trichterausschnitte und 2700 Hintergrundausschnitte der Größe  $25 \times 25$  Pixel verwendet. Siehe hierzu auch Kapitel 3.3.2.

<sup>2</sup>engl.: Linear Discriminant Analysis (LDA)

<sup>3</sup>engl.: Principal Components Analysis (PCA)



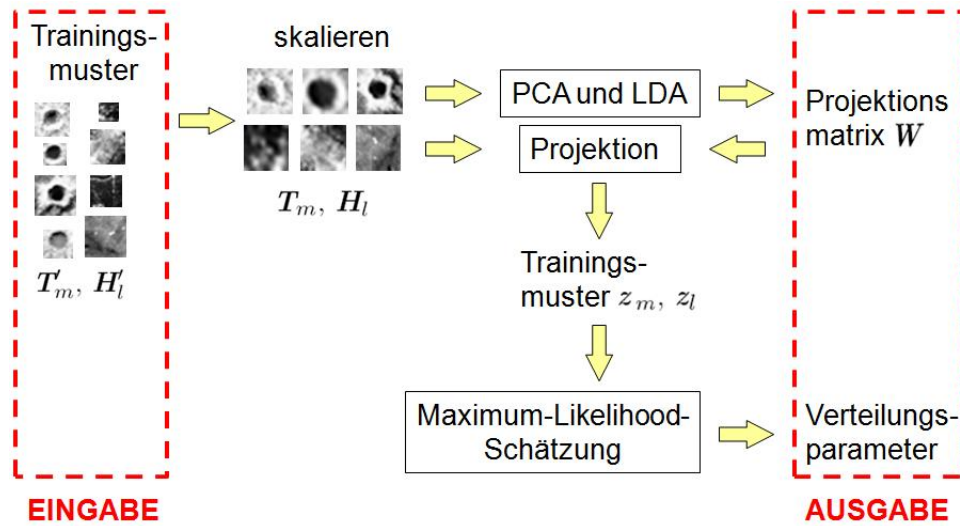


Abbildung 2.3: Ablauf der Bestimmung der Verteilungsparameter der Klassen Trichter und Hintergrund

nentenanalyse und  $\mu_x$  der Mittelwert aller Trainingsmuster  $x_n$  ist. Das Abziehen des Mittelwertes  $\mu_x$  sorgt für eine Translation des Koordinatensystems der Eingangsdaten  $\mathbf{X}$  in den Schwerpunkt der Eingangsdaten  $\mathbf{X}$ . Damit gilt für den Mittelwert  $\mu_y$  der projizierten Trainingsmuster:

$$\mu_y = \frac{1}{N} \sum_{n=1}^N y_n = 0$$

Die in den PCA-Raum projizierten Bildausschnitte  $y_n$  werden zu einer Matrix  $\mathbf{Y} = [y_1, \dots, y_N] \in \mathbb{R}^{K \times N}$  zusammengefasst.

Mit den projizierten Ausschnitten als Eingangsdaten  $\mathbf{Y}$  wird eine Fisher'sche Diskriminanzanalyse durchgeführt, die dazu dient, die Matrix  $\mathbf{V} = [v_1, \dots, v_R] \in \mathbb{R}^{K \times R}$  zu bestimmen, die die Daten so in den  $R$ -dimensionalen Raum projiziert, dass sie dort optimal nach ihrer Klasse getrennt werden können. Die Formel

$$z_n = \mathbf{V}^T \cdot y_n$$

beschreibt die Projektion. Da die  $y_n$ , wie oben erläutert, bereits auf den Schwerpunkt reduziert sind, ist hier das Abziehen des Mittelwertes  $\mu_y$  nicht mehr notwendig. Setzt man die obige Formel für  $y_n$  nun in diese ein, erhält man

$$z_n = \mathbf{V}^T \mathbf{U}_K^T \cdot (x_n - \mu_x) = \mathbf{W}^T \cdot (x_n - \mu_x)$$

Wir können die in den  $R$ -dimensionalen Raum projizierten Trainingsmuster  $z_n$  also auch direkt aus den Eingangsdaten  $x_n$  mit Hilfe einer Matrix  $\mathbf{W}^T = \mathbf{V}^T \mathbf{U}_K^T \in \mathbb{R}^{K \times D}$  berechnen.

Durch Projektion der Eingangsdaten  $\mathbf{X}$ , die aus den  $t_m$  und  $h_l$  bestehen, mit der Matrix  $\mathbf{W}^T$ , erhalten wir die in den  $R$ -dimensionalen Raum projizierten Trichterausschnitte  $z_m$  und Hintergrundausschnitte  $z_l$ . Aus der Menge der  $z_m$  und der Menge der  $z_l$  können nun Parameter von zwei noch zu bestimmenden Verteilungen (siehe hierzu Kapitel 3.4.3) mit der Maximum-Likelihood-Methode geschätzt werden. Mit Hilfe der Wahrscheinlichkeitsdichtefunktionen  $\mathcal{P}_T(\cdot)$  und  $\mathcal{P}_H(\cdot)$ , die sich aus der Schätzung ergeben, können in der Arbeitsphase die Kandidaten einer

Klasse zugeordnet werden. Abbildung 2.3 stellt den Ablauf der Dimensionsreduktion und der Bestimmung der Verteilungsparameter dar.

### 2.1.2 Arbeitsphase

Die Kandidatensuche liefert, wie bei der Auswahl der Trainingsmuster beschrieben, durch Kreuzkorrelation des Grauwertbilds  $\mathbf{B}$  mit den Templates  $\mathbf{M}_s$  die Bildfenster  $\mathbf{G}'_p$ . Diese werden wie die Trainingsausschnitte auf die Größe  $I \times J$  skaliert und wir erhalten als zu klassifizierende Kandidaten  $P$  Bildausschnitte  $\mathbf{G}_p \in \mathbb{R}^{I \times J}$ , oder als Merkmalsvektoren  $\mathbf{g}_p = \text{vec}(\mathbf{G}_p) \in \mathbb{R}^{D \times 1}$ . Ebenso wie die Trainingsausschnitte werden die Kandidaten  $\mathbf{g}_p$  mit der Projektionsmatrix  $\mathbf{W}^T$  in den  $R$ -dimensionalen Raum projiziert.

$$\bar{\mathbf{g}}_p = \mathbf{W}^T \cdot (\mathbf{g}_p - \boldsymbol{\mu}_x)$$

Für jedes  $\bar{\mathbf{g}}_p$  wird die Wahrscheinlichkeitsdichte  $\mathcal{P}_T(\bar{\mathbf{g}}_p)$  der Verteilung der Trichter und die Wahrscheinlichkeitsdichte  $\mathcal{P}_H(\bar{\mathbf{g}}_p)$  der Verteilung des Hintergrunds bestimmt. Ist  $\mathcal{P}_T(\bar{\mathbf{g}}_p) > \mathcal{P}_H(\bar{\mathbf{g}}_p)$ , so wird  $\bar{\mathbf{g}}_p$  der Klasse der Trichter  $\mathcal{T}$  zugeordnet, andernfalls der Klasse des Hintergrunds  $\mathcal{H}$ . Abbildung 2.4 stellt den Ablauf der Arbeitsphase dar.

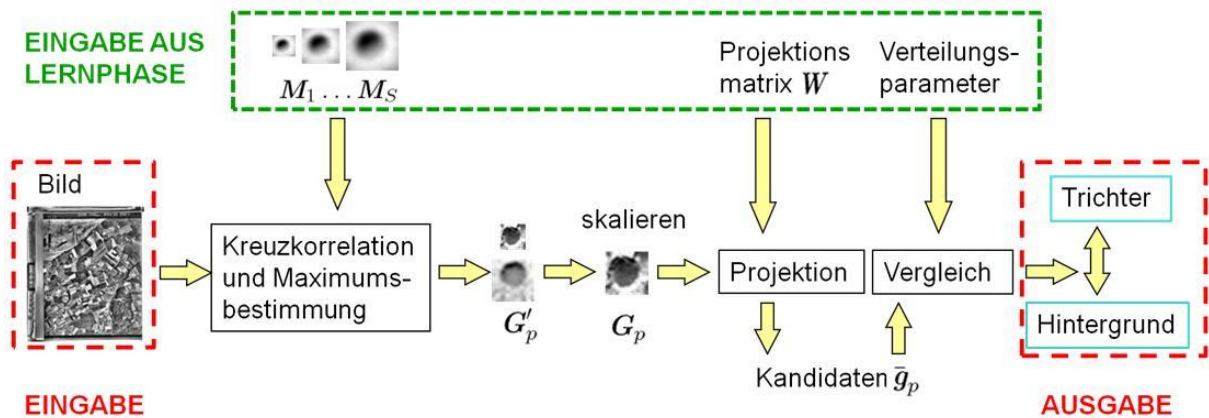


Abbildung 2.4: Ablauf der Arbeitsphase

### 2.1.3 Erweiterung des Konzepts

Das Konzept kann leicht auf mehrere Trichterklassen erweitert werden. Es ist etwa die Trennung nach mit Wasser gefüllten Trichtern, Trichtern mit großem Auswurf, nahezu runden Trichtern oder Trichtern, die aufgrund der Beleuchtungsverhältnisse eher sichelförmig aussehen, denkbar. Für jede dieser  $C$  Klassen müssten hierzu Trainingsbeispiele bei der Annotation gefunden werden. Dann könnten nach dem oben beschriebenen Ansatz insgesamt  $C$  Templates für die Kreuzkorrelation berechnet werden. Auch die Fisher'sche lineare Diskriminanzanalyse ist auf mehr als zwei Klassen erweiterbar, so dass die Projektion in einen Unterraum, in dem die Klassen gut trennbar sind, möglich ist.

Bei der Klassifikation sind bei mehreren Klassen mindestens zwei Ansätze denkbar: Beim ersten Ansatz könnten die Verteilungsparameter für jede Klasse bestimmt und die Kandidaten klassifiziert werden, indem sie der Klasse zugeordnet werden, bei der die Wahrscheinlichkeitsdichte

am größten ist. Damit würde die Bombentrichterdetektion als Erweiterung der Aufgabenstellung verschiedene Trichterklassen unterscheiden. Beim zweiten Ansatz könnte bei der Klassifikation auch weiterhin nur eine Trichterklasse angenommen werden, die z.B. durch eine gauss'sche Mischverteilung modelliert wird. Somit diene die Annotation mehrerer Trichterklassen zur Verfeinerung der Kandidatensuche und besseren Schätzung der Verteilung der Klasse der Trichter, nicht aber zur Unterscheidung verschiedener Trichterarten bei der Detektion.

## 2.2 Theoretische Grundlagen

Im oben erläuterten Konzept werden zwei Verfahren zur Dimensionsreduktion benötigt: Die Hauptkomponentenanalyse und die Fisher'sche lineare Diskriminanzanalyse. Die zum Verständnis des Konzepts wichtigen Aspekte der beiden Verfahren werden in den folgenden beiden Abschnitten dargestellt. Beide Verfahren werden bei Bishop [2006] ausführlich erläutert, die Hauptkomponentenanalyse auf den Seiten 561-570, Fisher'sche lineare Diskriminanzanalyse auf den Seiten 186-192.

### 2.2.1 Hauptkomponentenanalyse

Das Ziel bei der Hauptkomponentenanalyse ist es, eine Menge von  $N$  Punkten aus dem  $D$ -dimensionalen Raum so in einen  $K$ -dimensionalen Raum ( $K < D$ ) zu projizieren, dass möglichst wenig Information verloren geht. So kann man ursprünglich hochdimensionale Daten auf ihre wesentlichen Merkmale reduzieren. Die Idee ist nun, eine Hauptachsentransformation auf den Daten durchzuführen und sie in den von  $K$  Hauptachsen aufgespannten Raum zu projizieren. Abbildung 2.5 zeigt ein Beispiel für eine 2-dimensionale Punktwolke, die mit der ersten Hauptachse in den 1-dimensionalen Raum projiziert wird.

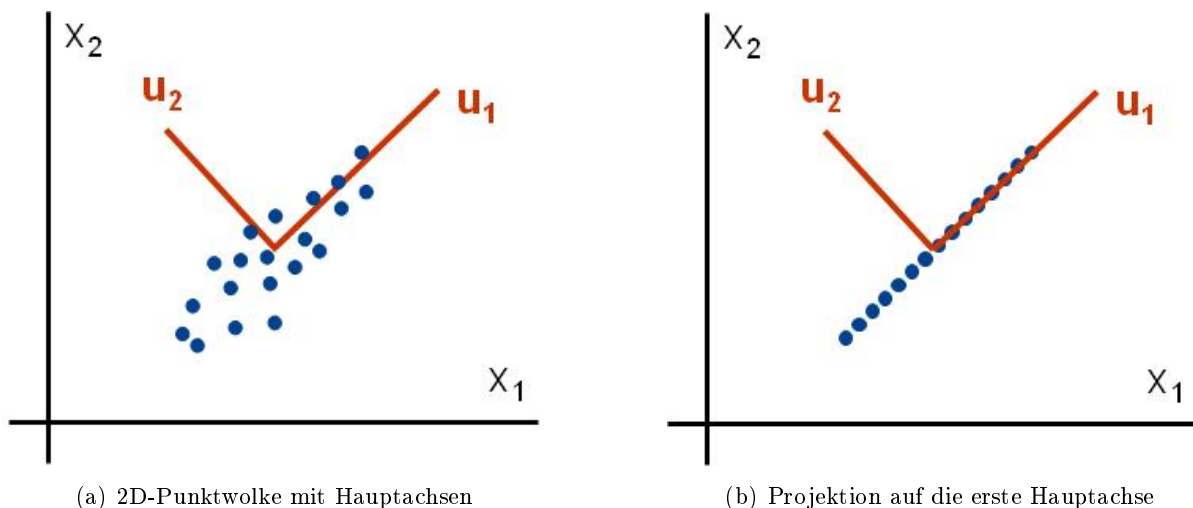


Abbildung 2.5: Projektion mit der Hauptkomponentenanalyse

Gehen wir zunächst von einer Projektion der  $\mathbf{x}_n$  ( $n = 1, \dots, N$ ) Punkte in den 1-dimensionalen Raum ( $K = 1$ ) aus. Die Gleichung zur Berechnung des projizierten Punktes  $y_n$  lautet dann:

$$y_n = \mathbf{u}_1^T \mathbf{x}_n$$

Hierbei ist der Vektor  $\mathbf{u}_1$  so zu bestimmen, dass die Varianz  $\sigma_y^2$  der projizierten Daten maximal wird. Dies gewährleistet einen minimalen Informationsverlust bei der Projektion.  $\sigma_y^2$  berechnet sich nach dem Varianzfortpflanzungsgesetz aus der Kovarianzmatrix  $\mathbf{S}$  der Daten  $\mathbf{x}_n$  und dem Vektor  $\mathbf{u}_1$ :

$$\sigma_y^2 = \mathbf{u}_1^T \mathbf{S} \mathbf{u}_1$$

Zur Berechnung benötigen wir also die Kovarianzmatrix  $\mathbf{S}$  der Punkte  $\mathbf{x}_n$ :

$$\mathbf{S} = \frac{1}{N} \sum_{n=1}^N (\mathbf{x}_n - \boldsymbol{\mu}_x)(\mathbf{x}_n - \boldsymbol{\mu}_x)^T \quad \text{mit} \quad \boldsymbol{\mu}_x = \frac{1}{N} \sum_{n=1}^N \mathbf{x}_n$$

$\boldsymbol{\mu}_x$  ist dabei der Mittelwert der Punkte  $\mathbf{x}_n$ . Da sich die Varianz  $\sigma_y^2$  mit der Länge von  $\mathbf{u}_1$  vergrößert und somit beliebig groß werden kann, führen wir eine Nebenbedingung ein:

$$\mathbf{u}_1^T \mathbf{u}_1 = 1$$

Die bezüglich  $\mathbf{u}_1$  zu maximierende Funktion lautet also:

$$\mathbf{u}_1^T \mathbf{S} \mathbf{u}_1 + \lambda_1 (1 - \mathbf{u}_1^T \mathbf{u}_1)$$

Nach Nullsetzen der ersten Ableitung ergibt sich die folgende Gleichung:

$$\mathbf{S} \mathbf{u}_1 = \lambda_1 \mathbf{u}_1$$

$\mathbf{u}_1$  muss also ein Eigenvektor von  $\mathbf{S}$  sein. Multipliziert man die Gleichung von links mit  $\mathbf{u}_1^T$  und nutzt die Nebenbedingung aus, so erhält man für die maximale Varianz  $\sigma_y^2$  der Daten den Wert  $\lambda_1$ .

$$\mathbf{u}_1^T \mathbf{S} \mathbf{u}_1 = \lambda_1 = \sigma_y^2$$

Offenbar wird  $\sigma_y^2$  dann maximal, wenn  $\mathbf{u}_1$  als der zum größten Eigenwert  $\lambda_1$  gehörige Eigenvektor der Kovarianzmatrix  $\mathbf{S}$  gewählt wird.  $\mathbf{u}_1$  wird als erste Hauptkomponente bezeichnet.

Für die Erweiterung auf die Projektion in den  $K$ -dimensionalen Raum wählen wir als nächste Hauptkomponente den Vektor  $\mathbf{u}_2$ , der die projizierte Varianz orthogonal zu  $\mathbf{u}_1$  maximiert. So verfahren wir für die weiteren  $\mathbf{u}_k$  ( $k = 3, \dots, K$ ) Vektoren. Als  $K$  Hauptkomponenten erhalten wir damit die zu den  $K$  größten Eigenwerten  $\lambda_1, \dots, \lambda_K$  gehörenden Eigenvektoren  $\mathbf{u}_1, \dots, \mathbf{u}_K$ . Die Projektion der Daten in den  $K$ -dimensionalen Unterraum ergibt sich dann mit den in einer Matrix  $\mathbf{U}_K$  zusammengefassten  $K$  Eigenvektoren nach der Formel

$$\mathbf{y}_n = \mathbf{U}_K^T \cdot (\mathbf{x}_n - \boldsymbol{\mu}_x) \quad \text{mit} \quad \mathbf{U}_K = [\mathbf{u}_1 \ \mathbf{u}_2 \ \dots \ \mathbf{u}_K]$$

Der Mittelwert  $\boldsymbol{\mu}_x$  wird deshalb von den Punkten  $\mathbf{x}_n$  abgezogen, damit der Schwerpunkt  $\boldsymbol{\mu}_y = \frac{1}{N} \sum_{n=1}^N \mathbf{y}_n$  der projizierten Punkte  $\mathbf{y}_n$  im Ursprung des Unterraums liegt.

### 2.2.2 Fisher's lineare Diskriminanzanalyse

Gegeben sind  $N$  Punkte  $\mathbf{y}_n$  im  $K$ -dimensionalen Raum.  $N_1$  Punkte gehören zur Klasse  $\mathcal{C}_1$  und  $N_2$  Punkte zur Klasse  $\mathcal{C}_2$ . Bei Fisher's linearer Diskriminanzanalyse ist ein Vektor  $\mathbf{v}$  gesucht, der die Punkte so in den 1-dimensionalen Raum projiziert, dass die beiden Klassen dort möglichst gut trennbar sind. Abbildung 2.6 stellt die Situation dar.

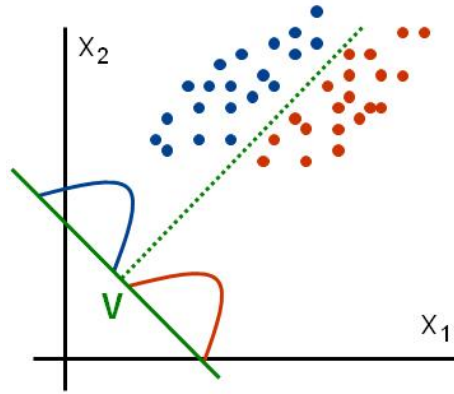


Abbildung 2.6: blau: Punkte der Klasse  $\mathcal{C}_1$  - rot: Punkte der Klasse  $\mathcal{C}_2$

Die Formel für die 1-dimensionalen projizierten Punkte  $z_n$  lautet also:

$$z_n = \mathbf{v}^T \mathbf{y}_n$$

Die Idee ist,  $\mathbf{v}$  so zu wählen, dass die Varianz zwischen den Klassen, also der Abstand der projizierten Klassenmittelpunkte, möglichst groß wird. Dabei soll die Varianz der projizierten Punkte innerhalb der einzelnen Klassen aber möglichst klein bleiben (siehe Abbildung 2.7).

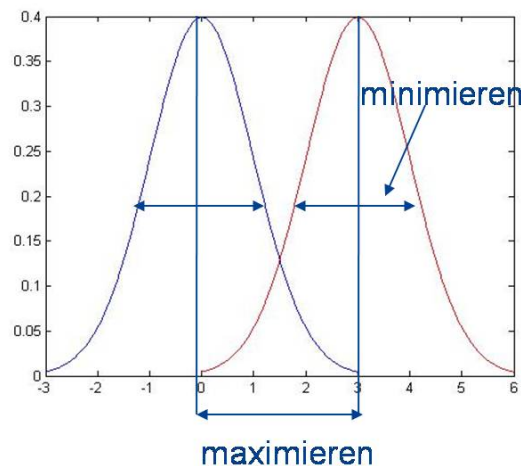


Abbildung 2.7: blau: Verteilung der Punkte in Klasse  $\mathcal{C}_1$  - rot: Verteilung der Punkte in Klasse  $\mathcal{C}_2$

Bishop [2006] zeigt, dass dies erfüllt ist, wenn das sog. Fisher-Kriterium  $J(\mathbf{v})$  bezüglich  $\mathbf{v}$  maximiert wird.

$$J(\mathbf{v}) = \frac{\mathbf{v}^T \mathbf{S}_B \mathbf{v}}{\mathbf{v}^T \mathbf{S}_W \mathbf{v}}$$

$\mathbf{S}_B$  ist dabei die Kovarianzmatrix zwischen den Klassen

$$\mathbf{S}_B = (\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)^T$$

und  $\mathbf{S}_W$  die Kovarianzmatrix innerhalb der Klassen.

$$\mathbf{S}_W = \sum_{n \in \mathcal{C}_1} (\mathbf{y}_n - \boldsymbol{\mu}_1)(\mathbf{y}_n - \boldsymbol{\mu}_1)^T + \sum_{n \in \mathcal{C}_2} (\mathbf{y}_n - \boldsymbol{\mu}_2)(\mathbf{y}_n - \boldsymbol{\mu}_2)^T$$

$\boldsymbol{\mu}_1$  und  $\boldsymbol{\mu}_2$  sind die Klassenmittelpunkte.

$$\boldsymbol{\mu}_1 = \frac{1}{N_1} \sum_{n \in \mathcal{C}_1} \mathbf{y}_n \quad \boldsymbol{\mu}_2 = \frac{1}{N_2} \sum_{n \in \mathcal{C}_2} \mathbf{y}_n$$

Maximiert man das Fisher-Kriterium, ergibt sich als Lösung für  $\mathbf{v}$ :

$$\mathbf{v} = \mathbf{S}_W^{-1}(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)$$

Die Fisher'sche Diskriminanzanalyse ist erweiterbar auf mehrere Klassen. Die Punkte  $\mathbf{y}_n$  werden dann nicht in den 1-dimensionalen Raum, sondern in einen  $R$ -dimensionalen Raum projiziert, wobei  $R = C - 1$  und  $C$  die Anzahl der Klassen  $\mathcal{C}$  ist.

Statt des Gewichtsvektors  $\mathbf{v}$ , mit dem die Punkte in den 1-dimensionalen Raum projiziert werden können, suchen wir nun eine Matrix  $\mathbf{V} = [\mathbf{v}_1, \dots, \mathbf{v}_R]$  mit den Gewichtsvektoren  $\mathbf{v}_1 \dots \mathbf{v}_R$ , so dass gilt

$$\mathbf{z}_n = \mathbf{V}^T \cdot \mathbf{y}_n$$

Wieder soll die Varianz der projizierten Punkte  $\mathbf{z}_n$  zwischen den Klassen maximiert, und innerhalb der Klassen minimiert werden. Es gibt nun diverse Möglichkeiten, das Fisher-Kriterium  $J(\mathbf{V})$  zu definieren. Eine Möglichkeit ist die folgende Form [Fukunaga, 1990]

$$J(\mathbf{V}) = \text{Tr} \{ (\mathbf{V} \mathbf{S}_W \mathbf{V}^T) (\mathbf{V} \mathbf{S}_B \mathbf{V}^T) \}$$

Diesmal sind  $\mathbf{S}_B$  und  $\mathbf{S}_W$  definiert als

$$\begin{aligned} \mathbf{S}_W &= \sum_{c=1}^C \sum_{n \in \mathcal{C}_c} (\mathbf{y}_n - \boldsymbol{\mu}_c)(\mathbf{y}_n - \boldsymbol{\mu}_c)^T & \text{mit} \quad \boldsymbol{\mu}_c &= \frac{1}{N_c} \sum_{n \in \mathcal{C}_c} \mathbf{y}_n \\ \mathbf{S}_B &= \sum_{c=1}^C N_c (\boldsymbol{\mu}_c - \boldsymbol{\mu}_y)(\boldsymbol{\mu}_c - \boldsymbol{\mu}_y)^T & \text{mit} \quad \boldsymbol{\mu}_y &= \frac{1}{N} \sum_{n=1}^N \mathbf{y}_n \end{aligned}$$

$N_c$  ( $c = 1, \dots, C$ ) ist hierbei die Anzahl der Punkte in der Klasse  $\mathcal{C}_c$ . Fukunaga [1990] zeigt, dass  $J(\mathbf{V})$  maximiert wird, wenn für die Spalten von  $\mathbf{V}$  die Eigenvektoren gewählt werden, die zu den  $K$  größten Eigenwerten der Matrix  $\mathbf{S}_W^{-1} \mathbf{S}_B$  gehören. Da die Matrix  $\mathbf{S}_B$  aus der Summe von  $C$  dyadischen Produkten mit dem Rang 1 gebildet wird, und zusätzlich noch die Bedingung  $\boldsymbol{\mu}_y = \frac{1}{N} \sum_{n=1}^N \mathbf{y}_n$  gilt, kann  $\mathbf{S}_B$  höchstens den Rang  $C - 1$  haben. Daher sind auch höchstens  $C - 1$  Eigenwerte der Matrix  $\mathbf{S}_W^{-1} \mathbf{S}_B$  ungleich Null. Es ist deshalb nicht sinnvoll, die Dimension  $R$  größer als die um eins reduzierte Anzahl der Klassen zu wählen.

# 3. Umsetzung

## Implementierung und Parameterwahl

In diesem Kapitel wird beschrieben, wie das in Kapitel 2 aufgestellte Konzept umgesetzt wurde. Hierzu wird zunächst auf das zur Verfügung stehende Bildmaterial und die Vorarbeiten, die zur Umsetzung nötig sind, eingegangen. Danach wollen wir die Vorgehensweise bei der Erstellung des Templates und bei der Kandidatensuche betrachten. Anschließend wird die Implementierung und Parameterwahl bei der Erstellung der Trainingsmuster und dem Training selbst erläutert. Da zum Test des Algorithmus auch die Trainingsmuster klassifiziert werden, wird in diesem Zusammenhang auch auf die Realisierung der Klassifizierung eingegangen. Den Abschluss des Kapitels bildet ein Abschnitt über die Ausschnittsgröße der Trainingsmuster, die einen Einfluss auf die Klassifikation hat.

### 3.1 Bildmaterial und Vorarbeiten

#### 3.1.1 Vorliegendes Bildmaterial

Die insgesamt etwa 324000 Luftbilder, die dem Kampfmittelbeseitigungsdienst zur Verfügung stehen, wurden seit 2006 sukzessive eingescannt. Derzeit liegen ca. 120.000 Bilder in digitaler Form vor. Aus der großen Anzahl und der hohen Scan-Auflösung von  $21\mu\text{m}$  ergeben sich enorme Datenmengen. Ein Bild, in der vollen Auflösung mit einer Farbtiefe von 12 Bit gescannt, hat eine Größe von etwa  $11000 \times 12000$  Pixeln und benötigt durchschnittlich ca. 230 MB Speicherplatz. Viele Bildverarbeitungsprogramme haben bei dieser Bildgröße eine lange Ladezeit.

Um das schnelle Durchblättern der Bilder dennoch zu ermöglichen, wurden alle 324000 Bilder bereits in einer geringeren Auflösung eingescannt. Sie haben dann eine Größe von ca.  $4400 \times 4500$  Pixeln, und belegen jeweils einen Speicherplatz von 300 - 500 KB. Diese Übersichtsbilder sind in 4210 Ordner mit jeweils 77 Bildern aufgeteilt und benötigen insgesamt 120 GB Speicherplatz. Mit Hilfe der Übersichtsbilder ist es möglich, zunächst eine Auswahl an Bildern zu treffen, die für die aktuelle Aufgabe relevant sind, und diese im Anschluss in der vollen Auflösung aus der Datenbank zu laden. Auf dem Server, der dem Kampfmittelbeseitigungsdienst der Bezirksregierung Arnsberg mit Sitz in Hagen zur Verfügung steht, sind derzeit etwa ein Sechstel der hochauflösten Bilder gespeichert. Für die vorliegende Arbeit wurden 54 Bilder mit Hilfe der Übersichtsbilder ausgewählt und in hochauflöster Form von diesem Server kopiert. Sie wurden

in zwei Mengen aufgeteilt:

- Trainingsmenge: bestehend aus 24 Luftbildern (mit und ohne Trichter) aus dem ländlichen Raum, aus denen die Bildausschnitte für die Lernphase genommen werden, und mit denen die Arbeitsphase durchgeführt wird, um den Algorithmus zu testen.
- Testmenge: bestehend aus 30 Luftbildern (mit und ohne Trichter) aus dem ländlichen Raum, die zum Testen des Algorithmus mit Bildern, die denen der Trainingsmenge ähneln, dienen.

### 3.1.2 Verwendete Software

Das Programm wurde in Matlab implementiert. Matlab ermöglicht das einfache Einlesen und schnelle Operieren mit Bildmatrizen und ist deshalb für die Aufgabe gut geeignet. Bei der Umsetzung wurde die an der technischen Universität in Prag entwickelte STPR-Toolbox<sup>1</sup> für Matlab verwendet. Dort sind u.a. die Hauptkomponentenanalyse, Fisher's lineare Diskriminanzanalyse sowie verschiedene Funktionen zur Schätzung von Verteilungen implementiert.

### 3.1.3 Vorverarbeitung des Bildmaterials

Aus Zeit- und Effizienzgründen wurde beim Einscannen der Luftbilder ein fester Parametersatz verwendet [Drauschke et al., 2007]. Deshalb ist die Bildqualität des Scans zunächst oft schlecht. Die Bilder sind sehr hell, auf vielen ist augenscheinlich nichts zu erkennen. Für die hochauflösten Bilder wurde von Drauschke et al. [2007] ein spezielles Dodging-Verfahren zur Qualitätsverbesserung entwickelt. Das Bild wird dabei in Patches aufgeteilt und auf jedem eine Histogrammverebnung durchgeführt. Danach werden die durch die Verebnung entstandenen Kanten zwischen den Patches entfernt, so dass dabei aber die Kanten von Bildobjekten erhalten bleiben. Abbildung 3.1 zeigt ein Bild vor der Anwendung des Dodging-Verfahrens und danach. Das Dodging wurde als erster Arbeitsschritt auf alle für die Bachelorarbeit zur Verfügung stehen-

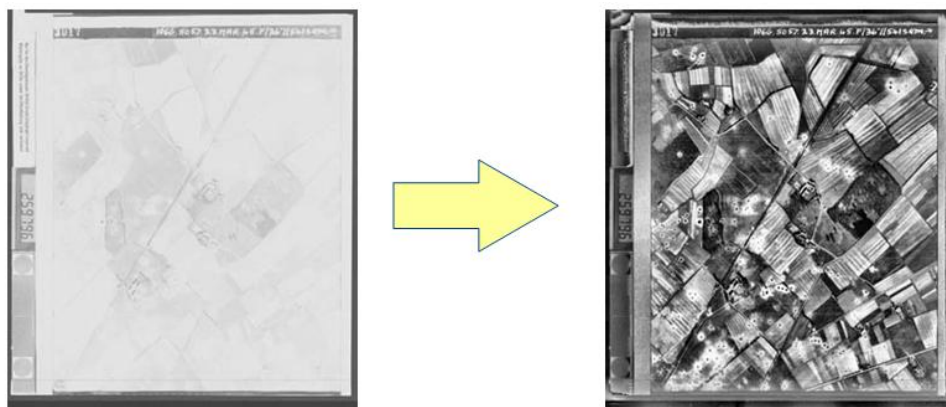


Abbildung 3.1: Ein Luftbild vor der Anwendung des Dodgings (links) und danach (rechts).

den Bilder angewandt. Die gedodgten Bilder wurden mit einer Farbtiefe von 8 Bit abgespeichert.

---

<sup>1</sup>STPR - Statistical Pattern Recognition



Da es aufgrund der Größe der Bilder bereits beim Einlesen in Matlab Speicherplatzprobleme gab, wurden sie von ca.  $11000 \times 12000$  Pixel auf ca.  $2000 \times 2300$  Pixel verkleinert. Dies gewährleistet, dass die Bilder nicht nur eingelesen werden können, sondern auch bei den Berechnungen die Speicherkapazität nicht überschritten wird.

Für die Bachelorarbeit wurden keine Vorgaben bezüglich der Laufzeit des Programms zur Detektion der Bombentrichter gemacht, dennoch ist es für Testzwecke wünschenswert, dass sich die Rechenzeit auf einen akzeptablen Zeitraum begrenzt. Auch dies ist durch die neue Bildgröße erfüllt. Das Verkleinern der Bilder könnte vermieden werden, wenn man die Trichterdetektion auf Bildausschnitten etwa mit einer Größe von  $2000 \times 2300$  Pixeln durchführt. Die Bombentrichter haben bei der hohen Auflösung aber eine zu unterschiedliche Grauwertverteilung, um mit einem einzigen allgemeinen Template eine genügend hohe Detektionsrate bei der Kandidatensuche zu erzielen. Bei der Verwendung mehrerer speziellerer Templates kann es später durchaus sinnvoll sein, auf Teilen von hochauflösten Bildern zu arbeiten.

### 3.1.4 Annotation der Trainingsbilder

Für die Erstellung des Templates  $M$  benötigen wir Bildausschnitte mit Bombentrichtern. Diese werden aus den 24 Trainingsbildern ausgeschnitten. Hierfür müssen in diesen zunächst die Positionen bekannt sein, an denen sich die Trichter befinden. Außerdem ist es bei der Auswahl der Trainingsmuster für die Klassifikation notwendig, die Trichterpositionen zu kennen, um diese mit den Positionen der gefundenen Kandidaten vergleichen zu können. Deshalb wurde ein Programm geschrieben, das die Markierung (Annotation) von Bombentrichtern in Bildern ermöglicht. Es werden dabei für jeden Trichter die Bildkoordinaten seiner Mitte und sein Radius bestimmt. Die Kenntnis des Radius ist notwendig, da er später die Größe der auszuschneidenden Bildfenster bestimmt.

Das Zoomen im Bild ist in Matlab bei gleichzeitiger Nutzung der Funktion zur Markierung von Bildpunkten nicht möglich. Deshalb zeigt das Programm das aktuelle Bild für die Annotation in Teilausschnitten an. Die Größe (Anzahl der angezeigten Zeilen und Spalten der Bildmatrix) des Ausschnitts kann der Benutzer verändern. Mit dieser Angabe ist die Betrachtung des Bildes in unterschiedlichen Vergrößerungsstufen möglich, da die Größe des Anzeigefensters bei jeder Zeilen-/Spaltenanzahl gleich bleibt. Die Ausschnitte decken das gesamte Bild zeilenweise ab und überlappen sich jeweils mit einer vom Benutzer wählbaren Pixelanzahl in Quer- und in Längsrichtung. Trichter, die sich zunächst nur teilweise im Ausschnitt befinden, sind im nächsten Ausschnitt durch die Überlappung ganz zu sehen und sicher zu markieren.

Zur Markierung eines Bombentrichters im Bild klickt der Benutzer jeweils in die Mitte des Trichters und an den äußeren Rand. Die angeklickten Bildkoordinaten liegen dann zunächst als Gleitkommazahlen vor. Durch Bestimmung des euklidischen Abstandes zwischen den beiden Punkten wird der Radius des Trichters ermittelt. Nun werden die Koordinaten der Trichtermitte  $(u_t, v_t)$  (Zeilennummer, Spaltennummer) und der Trichterradius  $r_t$  auf ganze Pixel gerundet und abgespeichert. Um zum nächsten Bildausschnitt zu gelangen muss der Benutzer die Return-Taste drücken.

Mit diesem Programm wurden in den 24 Luftbildern der Trainingsmenge insgesamt 1313 Bombentrichter markiert. Es war dabei aus Zeitgründen nicht möglich, in jedem Bild alle Trichter

zu annotieren; manche Bilder enthalten schätzungsweise 3000 Trichter und mehr. Da es später sowohl bei der Erstellung der Trainingsdaten für die Trainingsmuster des Hintergrunds als auch zum Teil bei der Bewertung der Klassifikationsergebnisse wichtig ist, alle Trichterpositionen in einem Bild zu kennen, werden die Bilder der Trainingsmenge in zwei Gruppen aufgeteilt:

- Gruppe 1 enthält die Bilder (16 Stück), auf denen alle Trichter annotiert wurden.
- Gruppe 2 enthält alle 24 Trainingbilder, inkl. derjenigen, auf denen nicht alle Trichter annotiert wurden.

Die Tabelle 3.1 zeigt einige statistische Daten zu den annotierten Trichtern. Die Spalte 2 gibt darüber Auskunft, ob auf dem Bild alle Trichter markiert sind, in Spalte 3 sind die Anzahlen der annotierten Trichter aufgelistet. In Spalte 4 und 5 ist der Mittelwert und die empirische Standardabweichung der Durchmesser<sup>2</sup> der Trichter im jeweiligen Bild angegeben. Die letzte Zeile der Tabelle 3.1 zeigt die Summe der annotierten Trichter, den Mittelwert aller mittleren Radien sowie den Mittelwert der Standardabweichungen.

Bildnr.	alle T mark.	# T	Mittl. Radius [pel]	Emp. Std.abw. [pel]
255515	ja	0		
259795		155	15.17	3.43
259865		282	14.17	3.60
261349	ja	7	14.14	1.58
266763	ja	0		
266791		61	15.97	3.91
266797		126	15.86	4.68
266820		36	17.39	2.41
266877		138	16.89	3.08
266882	ja	143	15.76	3.57
266892	ja	0		
267014	ja	19	12.40	2.87
267020	ja	12	16.85	3.61
268510	ja	18	15.89	4.13
268536	ja	13	14.86	3.46
268847	ja	0		
268853	ja	11	20.64	4.20
268857	ja	16	13.24	2.74
268859	ja	26	14.38	3.21
268861		81	12.60	3.45
271615	ja	0		
274840	ja	0		
277329	ja	52	19.42	3.77
277412		117	16.34	5.75
		1313	15.66	3.53

Tabelle 3.1: Statistik über die annotierten Bombentrichter

<sup>2</sup>Der Durchmesser des Trichters in Pixeln ergibt sich, wenn wir ausgehend von dem mittleren Pixel des Trichters jeweils nach rechts und nach links den Radius (in Pixeln) abtragen, um zum Rand des Trichters zu gelangen. Da wir nur mit ganzen Pixelzahlen arbeiten, berechnet sich der Durchmesser daher nach  $2 \cdot r_t + 1$ . Es ergeben sich somit nur ungerade Trichterdurchmesser. Der Mittelpunkt eines Trichters ist dadurch aber eindeutig festgelegt.

Die Kreuzkorrelation zur Kandidatensuche soll mit einem Template in verschiedenen Größen erfolgen. Um die zu verwendenden Templategrößen ermitteln zu können, wurde ein Histogramm über die Größen aller Trichterausschnitte gebildet, das einen Überblick über die Größenverteilung der annotierten Trichter auf den Trainingsbildern gibt. In der Abbildung 3.2 erkennt man, dass die kleinsten annotierten Trichter einen Durchmesser von 5 Pixeln und die größten einen Durchmesser von 35 Pixeln haben. Die meisten annotierten Bombentrichter haben einen Durchmesser von 17 Pixeln.

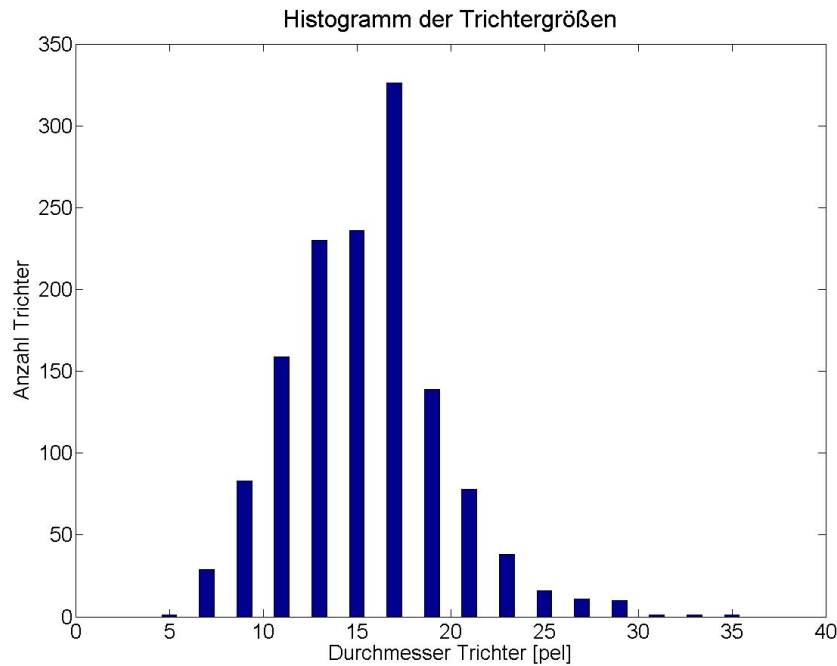


Abbildung 3.2: Die Durchmesser (in Pixel) aller annotierten Bombentrichter in 24 Bildern

### 3.1.5 Ausschneiden und Skalieren von Bildfenstern

Im Algorithmus müssen an verschiedenen Stellen Bildfenster ausgeschnitten, und auf die Größe  $I \times J$  skaliert werden. In der Lernphase ist dies bei der Erstellung des Templates und der Auswahl der Trainingsmuster erforderlich. In der Arbeitsphase müssen die gefundenen Kandidaten aus dem Bild ausgeschnitten und zur Klassifikation ebenfalls skaliert werden. Da die Bombentrichter im Wesentlichen rund sind, bietet es sich an, jeweils quadratische Bildfenster zu verwenden. Da  $I$  und  $J$  dann gleich sind, wollen wir im Folgenden für die Größe der Bildfenster nur eine Zahl  $I$  angeben, die der Zeilen- und gleichzeitig der Spaltenanzahl des Bildfensters entspricht.

Zum Ausschneiden und Skalieren von Bildfenstern wurde eine Funktion `bildausschnitte` geschrieben, die als Eingabe drei Parameter erwartet. Der erste Parameter ist das Bild, aus dem die Ausschnitte genommen werden sollen. Der zweite Parameter ist eine  $P \times 3$ -Matrix, die in der ersten und zweiten Spalte die Zeilen- und Spaltennummern der Mittelpunkte  $(u_p, v_p)$  der zu nehmenden Bildausschnitte, und in der dritten Spalte jeweils die halbe Ausschnittsbreite  $r_p$  enthält. Der dritte Parameter ist die Größe  $I$ , auf die die ausgeschnittenen Bildfenster skaliert werden sollen. Die Ausschnitte  $\mathbf{A}'_p$  werden aus dem Bild um die Mittelpunkte  $(u_p, v_p)$  herum

ausgeschnitten

$$\mathbf{A}'_p = \mathbf{B}(u_p - r_p : u_p + r_p, v_p - r_p : v_p + r_p)$$

Man erhält also Bildausschnitte mit einer ungeraden Zeilen- und Spaltenzahl von jeweils  $2 \cdot r_p + 1$ . Liegt der Mittelpunkt eines Bildausschnittes innerhalb eines Streifens von 32 Pixeln Breite entlang des Bildrandes, so wird an dieser Stelle kein Ausschnitt entnommen. Der Wert wurde aus folgendem Grund nicht variabel in Abhängigkeit von  $r_p$  gestaltet: Es kann dann nicht gewährleistet werden, dass bei der mehrfachen Ausführung der Funktion `bildausschnitte` mit denselben Mittelpunkten  $(u_p, v_p)$ , aber unterschiedlichen Größen  $r_p$ , jeweils exakt dieselbe Anzahl an Ausschnitten aus einem Bild vorliegt. Dies ist aber für Testzwecke notwendig. Der Wert von 32 Pixeln wurde deshalb gewählt, weil die größten während der Implementierung verwendeten Ausschnitte eine halbe Ausschnittsbreite von 32 Pixeln haben, und somit sichergestellt ist, dass bei der Ausschnittsentnahme nicht auf Bereiche außerhalb des Bildes zugegriffen wird.

Die verschieden großen  $\mathbf{A}'_p$  werden nicht direkt gespeichert, sondern zunächst auf die Größe  $I \times I$  skaliert. Für die Skalierung wird die Matlab-Funktion `imresize` mit bilinearer Interpolation verwendet. Es ist darauf zu achten, dass die Skalierung mit Bildmatrizen erfolgt, deren Einträge den Datentyp `uint8` haben. Erfolgt die Skalierung mit Matrizen, die bereits in das `double`-Format umgewandelt wurden, ergeben sich Probleme bei der Interpolation<sup>3</sup>.

Die Einträge der skalierten Bildmatrizen werden für die anschließenden Berechnungen vom `uint8`-Format in das `double`-Format umgewandelt. Da die Ausschnitte in den meisten Fällen (außer bei der Templateerstellung) nicht in Matrixform, sondern in Vektorform gebraucht werden, wird nun der `vec`-Operator auf die Bildausschnitte angewandt. So können sie auch in einer einfachen Datenstruktur gespeichert werden, nämlich einer 2-dimensionalen Matrix mit den vektorisierten Bildausschnitten als Spaltenvektoren. Diese Matrix ist der Rückgabeparameter der Funktion `bildausschnitte`.

## 3.2 Kandidatensuche

### 3.2.1 Erstellung des Templates

Die Bildfenster  $\mathbf{T}_t$  mit Trichtern werden mit der Funktion `bildausschnitte` ausgeschnitten. Als Ausschnittsmitten werden die annotierten Trichtermitten und als halbe Ausschnittsbreite der bei der Annotation festgelegte gerundete Trichterradius übergeben. Es stellt sich die Frage, auf welche Größe  $I$  die Trichterausschnitte zu skalieren sind. Bei der Verkleinerung von Bildausschnitten geht Information verloren, bei ihrer Vergrößerung wird durch Interpolation angenommene Information hinzugefügt. Weder Informationsverlust noch ungesicherter Informationsgewinn ist bei der Erstellung des Templates erwünscht. Damit keines der beiden Phänomene überwiegt, sollen bei der Skalierung etwa gleich viele Ausschnitte verkleinert wie vergrößert werden. Um außerdem möglichst wenige Ausschnitte skalieren zu müssen, wird als Einheitsgröße  $I$  für alle Trichterausschnitte die Größe von 17 Pixeln gewählt, die der Größe der meisten annotierten

<sup>3</sup>Bei der Implementierung wurde bei der Skalierung zunächst mit Matrizen im `double`-Format gearbeitet. Bei den Ausschnitten eines Bildes wurden diese fehlerhaft abgespeichert, so dass sie nicht mehr geladen werden konnten. Bei der Analyse ergab sich, dass dieser Fehler nicht auftritt, wenn eine Nearest-Neighbour-Interpolation oder das `uint8`-Format verwendet wird.

Trichter entspricht (Abbildung 3.2).

Das Muster  $\mathbf{M}$ , das als Template für die Kreuzkorrelation genutzt wird, wurde, wie in Kapitel 2.1.1 beschrieben, als Mittelwert aller 1313 Trichterausschnitte  $\mathbf{T}_t$  berechnet (Abbildung 3.3).

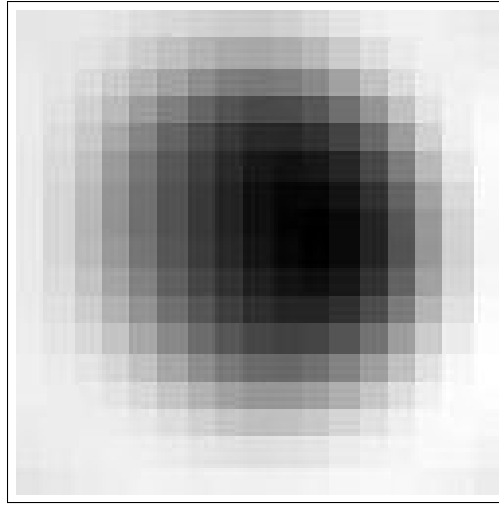


Abbildung 3.3: Mittlerer Trichter der Größe 17×17 Pixel

Als nächstes sind die zu verwendenden Templategrößen zu bestimmen. Da die meisten annotierten Trichter eine Größe von 17 Pixeln haben, wird diese als erste Templategröße festgelegt. Die Kreuzkorrelation kann mit Maßstabsdifferenzen von 20% gut umgehen. Die weiteren Größen wurden deshalb mit 33, 23, 11 und 9 festgelegt. In Tabelle 3.2 wurden die fünf Templategrößen jeweils einmal mit dem Faktor 1.2 und 0.8 multipliziert, um die maximale und minimale Größe der Trichter, die mit einem Template dieser Größe gefunden werden können, zu ermitteln. Die Spalte 3 zeigt das Ergebnis. Laut Tabelle 3.2 können mit den fünf Templategrößen alle Trichterdurchmesser im Bereich von 40 bis 7 Pixeln erfasst werden. Trichter, deren Durchmesser kleiner ist als 7 Pixel können also nicht gefunden werden. Da aber nur eine minimale Anzahl an Trichtern einen Durchmesser hat, der kleiner als 7 Pixel ist (siehe Abbildung 3.2), ist eine weitere kleinere Templategröße nicht notwendig.

Es werden nur ungerade Templategrößen verwendet. Der Grund ist, dass es sich so bei den Positionen der Maxima, die man bei der Kreuzkorrelation erhält, um klar definierte Bildausschnittsmitten handelt.

Größe	Faktor	
33	1.2	40
33	0.8	26
23	1.2	28
23	0.8	18
17	1.2	20
17	0.8	14
11	1.2	13
11	0.8	9
9	1.2	11
9	0.8	7

Tabelle 3.2: Minimale und maximale Trichtergrößen, die mit den Templates erreichbar sind

### 3.2.2 Kandidatensuche mittels Kreuzkorrelation

Für die Kreuzkorrelation wurde das Programm `korrelation_durchfuehren` geschrieben. Es skaliert das Template  $M$  (Abbildung 3.3) mit der Matlab-Funktion `imresize` auf die oben festgelegten Größen 9, 11, 17, 23 und 33. Dabei wird jeweils die bikubische Interpolation verwendet. Für jedes der fünf Templates wird nun zunächst die Kreuzkorrelation durchgeführt und dann die lokalen Maxima der Korrelationskoeffizienten-Matrix bestimmt und gespeichert.

Für die Kreuzkorrelation ruft das Programm die Matlab-Funktion `normxcorr2` auf, die als Eingabe das Template und das Bild erwartet. Die Funktion berechnet den Korrelationskoeffizienten zwischen dem Template und dem aktuellen Bildfenster an jeder Bildposition. Dies geschieht in Abhängigkeit von der Bildgröße entweder im Ortsraum, oder (bei großen Bildern) im Frequenzraum, was die Laufzeit verringert. Die zurückgelieferte Matrix  $K$  enthält `double`-Werte im Bereich -1.0 bis 1.0 und hat die um die Templategröße verminderte Größe des Ausgangsbildes  $U - I \times V - I$ . Das liegt daran, dass der Korrelationskoeffizient nur an den Bildpositionen berechnet werden kann, an denen das Template vollständig im Bild liegt.

Es müssen nun die lokalen Maxima der Matrix  $K$  bestimmt werden. Dies sind die Positionen, an denen das Template und das Bildfenster die größte Ähnlichkeit aufweisen, und somit möglicherweise Trichter sein könnten. Hierfür ruft das Programm die Funktion `positionen_maxima_sw`<sup>4</sup> auf. Als Eingabeparameter werden die Matrix  $K$ , die Bildgröße, die Templategröße, sowie ein Schwellwert  $\tau$  übergeben.  $\tau$  ist der Wert, den der Korrelationskoeffizient mindestens haben muss, um als lokales Maximum in  $K$  betrachtet zu werden.  $\tau$  wurde in der vorliegenden Arbeit auf den relativ niedrigen Wert 0.7 gesetzt, damit die Korrelation viele Kandidaten liefert und möglichst alle Trichter erfasst. Die Funktion `positionen_maxima_sw` liefert eine  $P \times 2$ -Matrix mit den Zeilen- und Spaltennummern im Bild, an denen sich in  $K$  lokale Maxima mit einem Wert  $\geq \tau$  befinden, zurück. Diese Matrix wird zusammen mit der zugehörigen halben Templategröße gespeichert. Sie enthält in den ersten beiden Spalten die Koordinaten der Ausschnittsmitten der durch Kreuzkorrelation ermittelten Kandidaten und in der dritten Spalte die Größe, mit der sie später aus dem Bild ausgeschnitten werden.

## 3.3 Erstellung der Trainingsmuster

### 3.3.1 Auswahl der Trainingsmuster

Als Trainingsmuster für die Bestimmung der Verteilungsparameter der Klassen Trichter und Hintergrund werden die Bildfenster verwendet, die bei der Kandidatensuche mittels Kreuzkorrelation auf den 24 Trainingsbildern ermittelt wurden.

Die Positionen der Kandidaten werden mit den annotierten Positionen auf den Trainingsbildern verglichen. Die Funktion `vergleich_mit_trainingsbild` teilt die in einem Bild bei der Kandidatensuche ermittelten und die annotierten Positionen in drei Kategorien ein:

- Kategorie 1 enthält die Positionen, die sowohl annotiert, als auch mit dem Template gefunden wurden.

---

<sup>4</sup>Die Funktion wurde von Dipl.-Ing. Susanne Wenzel (IGG Bonn, Professur für Photogrammetrie) erstellt.

- Kategorie 2 enthält die Positionen, die zwar annotiert, aber nicht vom Template erfasst wurden.
- Kategorie 3 enthält alle Positionen, die nicht annotiert, aber vom Template detektiert wurden.

Die Funktion `vergleich_mit_trainingsbild` wird für jedes Trainingsbild aufgerufen.

An allen Positionen der Kategorien 1 und 3 werden mit der Funktion `bildausschnitte` Bildfenster ausgeschnitten. Die Ausschnitte der Kategorie 1 verwenden wir als Trainingsmuster  $t_m$  der Klasse `Trichter` und die Ausschnitte der Kategorie 3 als Trainingsmuster  $h_l$  der Klasse `Hintergrund`. Bei den  $h_l$  handelt es sich nur um die Ausschnitte der Kategorie 3, die aus den Bildern der Gruppe 1 (siehe Abschnitt 3.1.4) stammen. Nur bei diesen Bildern kann man sicher sein, dass sich unter den detektierten, aber nicht annotierten Positionen keine Trichterpositionen befinden und somit die Trainingsmenge der Hintergrundausschnitte keine Trichterausschnitte enthält. Wir erhalten insgesamt 1629 Trainingsbeispiele für die Klasse `Trichter` und 6832 Trainingsbeispiele für die Klasse `Hintergrund`, die als Spaltenvektoren  $t_l$  und  $h_m$  gespeichert sind.

Die Anzahl aller annotierten Trichter ist 1313 (vgl. Tabelle 3.1). Die Anzahl der Trainingsbeispiele für Trichter ist mit 1629 größer, da ein und derselbe Trichter mit mehreren Templates gefunden werden kann. Die mehrfach gefundenen Ausschnitte werden alle als Trainingsmuster verwendet, weil auch später bei der Klassifikation der Kandidaten mehrfach gefundene Ausschnitte klassifiziert werden. Sonst könnte es passieren, dass Trichter, die bei einer Templategröße als solche erkannt werden würden, durch die vorherige Löschung dieses Ausschnittes nicht mehr gefunden werden können. Erst nach der Klassifikation werden dann mehrfach markierte Ausschnitte auf jeweils einen reduziert.

### 3.3.2 Anzahl und Verhältnis der Trainingsmuster

Die notwendige Hauptkomponentenanalyse mit den Trainingsmustern wird mit der Funktion `pca` durchgeführt. Sie ist in der STPR-Toolbox implementiert und von Susanne Wenzel für hochdimensionale Daten erweitert worden. Bei der Eingabe einer großen Anzahl von Trainingsmustern kommt es bei der Funktion `pca` zu Speicherplatzproblemen. Es können daher für die Lernphase nicht alle 8461 Trainingsmuster benutzt werden, sondern es muss eine Auswahl getroffen werden. Die maximal verwendbare Anzahl hängt von der Dimension  $D$  der Trainingsdaten und damit von der Ausschnittsgröße ab, und liegt bei der verwendeten Ausschnittgröße bei etwa 3600. Bei der Auswahl der zu verwendenden Trainingsmuster wird zunächst die Gesamtzahl der Trichterausschnitte  $M$  und der Hintergrundausschnitte  $L$  festgelegt. Dann wird eine zufällige Auswahl von  $M$  bzw.  $L$  Ausschnitten aus allen 1629 bzw. 6832 getroffen. Um zu gewährleisten, dass eine repräsentative Auswahl an Trainingsmustern aus allen Bildern für die Lernphase verwendet wird, werden die Anzahlen der zufällig zu wählenden Trichter- und Hintergrundausschnitte aus einem Bild festgelegt. Sie berechnet sich proportional zur Anzahl aller im Bild vorhandenen Trichter- bzw. Hintergrundausschnitte nach:

$$\text{Anzahl aller im Bild vorhandenen Trichterausschnitte} \cdot \frac{M}{1629}$$

$$\text{Anzahl aller im Bild vorhandenen Hintergrundausschnitte} \cdot \frac{L}{6832}$$

Aufgrund von Rundungsfehlern ergeben sich in der Summe allerdings meist nicht exakt  $M$  bzw.  $L$  Ausschnitte, sondern jeweils Abweichungen von  $\pm 1$  oder  $\pm 2$ .

Neben der Gesamtzahl der für das Training zu verwendenden Ausschnitte ist auch das Verhältnis von Trichter- zu Hintergrundausschnitten festzulegen. Beim Vergleich der Kandidaten mit den Trainingsbildern wurde festgestellt, dass das Verhältnis von Trichtern zu Hintergrund unter den zu klassifizierenden Kandidaten etwa 1:10 beträgt (vgl. Abschnitt 4.2.1). Es ist sinnvoll, das Training mit demselben Verhältnis der Beispiele durchzuführen, wie es bei den zu klassifizierenden Kandidaten erwartet wird. Würde man aber das Verhältnis 1:10 beim Training verwenden, läge die verwendete Trichteranzahl aufgrund der maximal zu verwendenden Gesamtzahl von 3600 Ausschnitten nur bei maximal 327. Diese geringe Anzahl ist ungünstig für die Bestimmung der Verteilungsparameter. Deshalb wurde für das Training ein Verhältnis von 1:3 gewählt. Dies berücksichtigt einerseits, dass im Normalfall mehr Hintergrund als Trichter gefunden wird, andererseits wird eine stabilere Bestimmung der Verteilungsparameter der Trichterklasse gewährleistet. Des Weiteren kann man vermuten, dass das Verhältnis von Trichter zu Hintergrund ohnehin stark schwankt, je nachdem welche Bilder verwendet werden, und wie viele Bombentrichter sie enthalten (vgl. Abschnitt 4.2.1).

Um die maximale Anzahl an Trainingsbeispielen von 3600 zu erreichen, geben wir also 900 Trichter- und 2700 Hintergrundausschnitte für das Training an. Verwendet werden (aufgrund von Rundungsfehlern) 900 und 2701. Die Ausschnittsgröße wird für das Training nicht mit 17, sondern mit 25 gewählt. Diese Wahl wurde bei empirischen Untersuchungen bei der Klassifikation ermittelt, und ist in Abschnitt 3.5 begründet. Die Dimension des betrachteten Merkmalsraums beträgt also 625. Alle ausgewählten Trichter- und Hintergrundausschnitte  $\mathbf{t}_m$  und  $\mathbf{h}_l$  werden in einer Matrix  $\mathbf{X} \in \mathbb{R}^{625 \times N}$  zusammengefasst.

## 3.4 Training

### 3.4.1 Hauptkomponentenanalyse

Es ist nun die Dimension  $K$  zu bestimmen, in welche die Trainingsdaten projiziert werden sollen. Dabei soll einerseits die Dimension möglichst stark reduziert werden, andererseits möglichst wenig Information verloren gehen. Wie in Abschnitt 2.2.1 erläutert, geben die Eigenwerte der Kovarianzmatrix  $\mathbf{S}$  der Daten jeweils ihre Varianz in Richtung des zugehörigen Eigenvektors an. Der Gesamtfehler, der durch die Projektion in einen  $K$ -dimensionalen Raum entsteht, kann also durch die Summe der verworfenen  $D - K$  Eigenwerte ausgedrückt werden. Abbildung 3.4 zeigt die Summe der verworfenen Eigenwerte in Abhängigkeit von der Dimension  $K$  des Unterraums. Es wurden hier bei der PCA 800 Trainingsausschnitte der Dimension  $25 \times 25$  verwendet. Der exponentiell abfallende Verlauf zeigt, dass auch bei einer (im Vergleich zu  $D$ ) kleinen Dimension  $K$  fast die gesamte Information der Daten erhalten bleibt. Mit Hilfe der Eigenwerte kann man also ein  $K$  bestimmen, bei dem der Gesamtfehler der Projektion kleiner oder gleich einem Prozentsatz der gesamten Varianz der Daten ist. Somit geht bei der Projektion in den  $K$ -dimensionalen Raum nur dieser Prozentsatz an Information verloren. Im verwendeten Beispiel (Abbildung 3.4) wird



z.B. die 5%-Schwelle bei der Dimension 41 unterschritten. Daher können mit Hilfe der PCA die Daten von der Dimension 625 ( $25 \times 25$ ) auf die Dimension 41 reduziert werden, wobei 95% der Information erhalten bleibt.

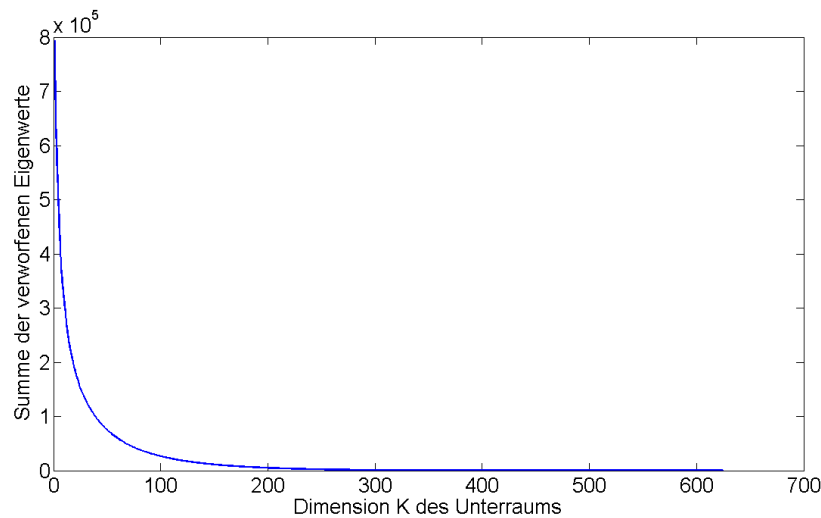


Abbildung 3.4: Summe der verworfenen Eigenwerte in Abhängigkeit von der Dimension  $K$ .

Die Funktion `pca` erwartet als Eingabe die Trainingsdaten  $\mathbf{X}$  in Spaltenvektorform. Als optionaler Parameter kann unter anderem der maximal zu tolerierende Informationsverlust (in Prozent), der bei der Projektion entsteht, angegeben werden. Wir geben hier einen maximalen Fehler bei der Projektion von 5% an.  $K$  wird von der Funktion dann so gewählt, dass diese Bedingung eingehalten ist. Die Funktion `pca` gibt ein Modell mit dem Datentyp `struct` zurück, das unter anderem die im Kapitel 2.1.1 beschriebene Matrix  $\mathbf{U}_K$  enthält, mit der die Daten in den  $K$ -dimensionalen Raum projiziert werden können. Außerdem liefert das Modell einen Vektor  $\mathbf{b} = -\mathbf{U}_K^T \cdot \boldsymbol{\mu}_x$ , den in den  $K$ -dimensionalen Unterraum projizierten negativen Mittelwert der Trainingsdaten. Dieser wird bei der Projektion benötigt.

Führt man die Hauptkomponentenanalyse mit 900 Trichter- und 2700 Hintergrundausschnitten der Größe  $25 \times 25$  durch, so wird die Dimension von 625 auf 45 reduziert<sup>5</sup>.

Nach der Hauptkomponentenanalyse werden die 625-dimensionalen Trainingsmuster mit der in der STPR-Toolbox implementierten Funktion `linproj` in den 45-dimensionalen Raum projiziert. Die Funktion `linproj` erwartet als Eingabe die zu projizierenden Daten  $\mathbf{X}$  als Spaltenvektoren, sowie ein Modell des Datentyps `struct`, wie es z.B. die Funktion `pca` liefert. Als Ausgabe erhalten wir die  $45 \times N$ -Matrix  $\mathbf{Y}$  mit den projizierten Trainingsmustern.

### 3.4.2 Fisher's lineare Diskriminanzanalyse

Nach der Projektion in den 45-dimensionalen Raum wird mit den Daten  $\mathbf{Y}$  die lineare Diskriminanzanalyse nach Fisher und danach die Projektion in den  $R$ -dimensionalen Raum durchgeführt. Nach Abschnitt 2.2.2 ist  $R = C - 1$  und  $C$  die Anzahl der Klassen. Da es sich in diesem Fall um zwei Klassen (Trichter und Hintergrund) handelt, werden die Daten also in den 1-dimensionalen Raum projiziert und dort ihre Verteilungen bestimmt.

<sup>5</sup>Die Dimension, auf die reduziert wird, schwankt, je nachdem welche Ausschnitte bei der Zufallswahl gewählt werden. Bei der letztendlich verwendeten Konfiguration liegt sie bei 45.

Für die LDA wird die ebenfalls in der STPR-Toolbox implementierte Funktion `lda` verwendet. Als Eingabe erwartet sie ein **struct** mit den Trainingsdaten als Spaltenvektoren in einer Matrix **Y**, sowie eine Angabe über die Klassenzugehörigkeit der Trainingsdaten in Form eines  $1 \times N$ -Zeilenvektors **l**. In dem Vektor **l** stehen Integer-Einträge von 1 bis  $C$ , die die Nummer der Klasse angeben, zu der das jeweilige Trainingsmuster gehört. Die Einträge in **l** bezeichnen wir als Labels der Trainingsmuster. **l** ist in diesem Fall also ein  $1 \times 3600$  Vektor mit 900 Einsen und 2700 Zweien. Optional kann bei der Funktion `lda` die Dimension  $R$  des Unterraums angegeben werden, in den die zurückgegebene Projektionsmatrix projiziert. Wird diese Angabe weggelassen, erhält der Unterraum die Dimension der Eingangsdaten  $K$ . In diesen Fall wird  $R$  auf 1 gesetzt. Die Funktion `lda` gibt, wie die Funktion `pca`, ein Modell des Datentyps **struct** zurück, das unter anderem die im Kapitel 2.1.1 erläuterte Matrix **V** enthält, mit der die Eingangsdaten in den  $R$ -dimensionalen Raum projiziert werden können, in dem sie gut trennbar sind. Hier ist **V** ein Spaltenvektor, da  $R = 1$  ist. Ein weiteres Element ist der Vektor **b**, der bei der Projektion benötigt wird und sich nach der Formel  $\mathbf{b} = -\mathbf{V}^T \cdot \boldsymbol{\mu}_y$  berechnet. In diesem Fall gilt  $\mathbf{b} = 0$ , da der Mittelwert der Eingangsdaten **Y** Null ist.

Die Trainingsmuster **Y** werden mit der oben erläuterten Funktion `linproj` zum Schätzen der Verteilung in den 1-dimensionalen Raum projiziert. Für die Bestimmung der Verteilungsparameter und die Klassifikation werden das PCA-Modell und das LDA-Modell, sowie die 1-dimensionalen Trainingsmuster mit ihren Labels gespeichert.

### 3.4.3 Schätzung der Parameter der Verteilungen

Im Kapitel 2.1 wurde dargelegt, dass die Parameter der Verteilungen der Trichter und des Hintergrunds aus den in den LDA-Raum projizierten Daten geschätzt werden müssen, um die mittels Kreuzkorrelation gefundenen Bombentrichter-Kandidaten klassifizieren zu können. Zunächst ist allerdings durch optische Betrachtung zu klären, zu welchen Wahrscheinlichkeitsverteilungen die Daten gehören. Würde man die 3600 Trainingsmuster im 1-dimensionalen Raum darstellen, so lägen sie alle auf einer Geraden übereinander, die Grafik wäre nicht interpretierbar. Deshalb wurde ein Histogramm der 1-dimensionalen Trainingsmuster erstellt (Abbildung 3.5). Dabei verwenden wir die Matlab-Funktion `hist`. Sowohl der Wertebereich der Trichter, als auch der des Hintergrunds wurde in 80 Abschnitte (bins) eingeteilt, um eine genügend feine Auflösung zu erhalten.

Abbildung 3.6 zeigt die Histogramme für die Klassen Trichter und Hintergrund getrennt. Es ist zu beachten, dass die Skala auf der y-Achse (Anzahl Trainingsmuster) nicht gleich ist, da dreimal so viele Hintergrund- wie Trichterausschnitte verwendet wurden. Da die Verteilungen in beiden Fällen einer Gauß-Kurve ähneln, werden im Rahmen dieser Arbeit als Verteilungen für beide Klassen Normalverteilungen angenommen.

Die Bestimmung der Parameter der Normalverteilung aus einer gegebenen Stichprobe ist einfach. Die beiden Momente (Erwartungswert und Varianz), die sie charakterisieren, lassen sich nach der Maximum-Likelihood-Methode mit dem Mittelwert und der empirischen Kovarianzmatrix der gegebenen Daten schätzen [Bishop, 2006]. Die Funktion `mlcgmm` in der STPR-Toolbox implementiert die Maximum-Likelihood-Schätzung. Als Eingabe erwartet sie die Daten der Stichprobe in Spaltenvektorform zusammengefasst zu einer Matrix. Sie liefert ein Modell des Daten-

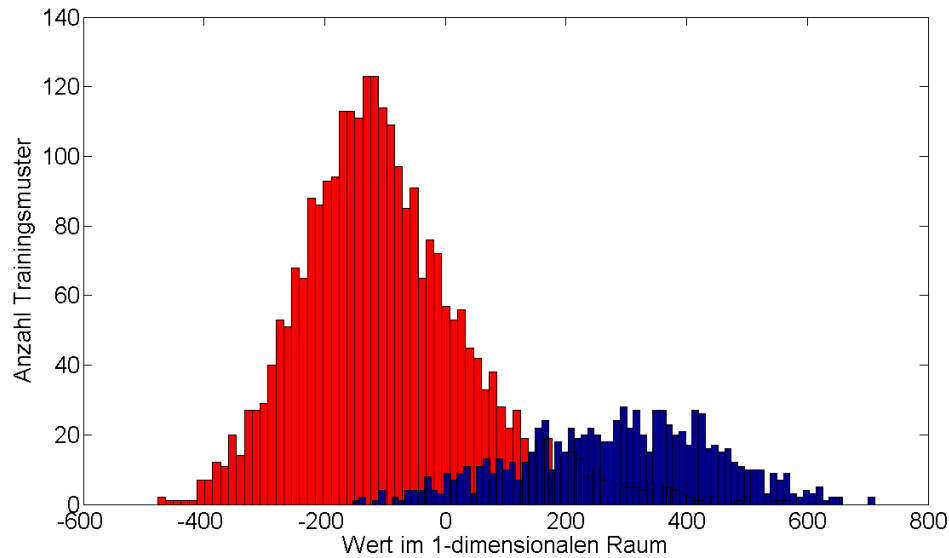


Abbildung 3.5: Histogramm der 3600 Bildausschnitte nach der LDA, rot: Hintergrund - blau: Trichter

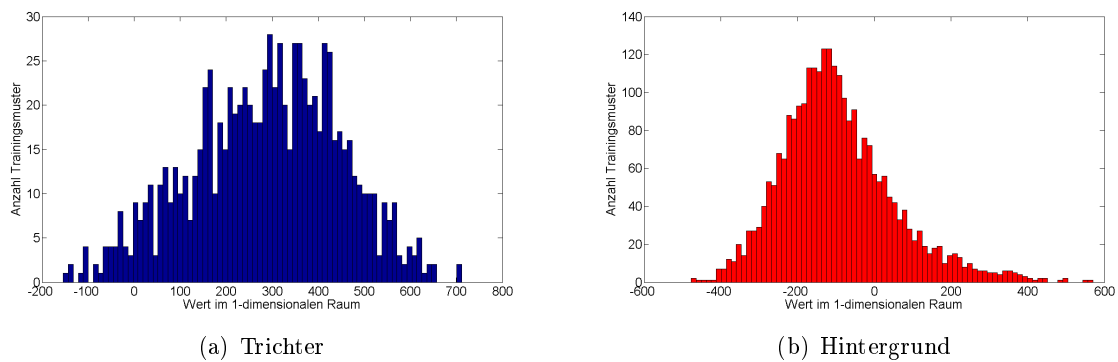


Abbildung 3.6: Histogramm der 900 Trichterausschnitte und 2700 Hintergrundausschnitte nach der LDA

typs `struct` zurück, das unter anderem den Mittelwert und die Kovarianzmatrix der Daten enthält. In diesem Fall handelt es sich um 1-dimensionale Eingabedaten, daher erhalten wir für beide Werte je ein Skalar. Die Funktion `mlcgmm` wurde je einmal für die 900 Trichterausschnitte und die 2700 Hintergrundausschnitte aufgerufen und die Modelle für die Klassifikation gespeichert.

### 3.4.4 Klassifizierung

In der Arbeitsphase und bei der Bestimmung der zu erwartenden Fehlerrate bei der Klassifikation müssen die in den LDA-Raum projizierten Daten klassifiziert werden. Die Klassifizierung erfolgt wie im Konzept (Abschnitt 2.1.2) beschrieben. Für die Bestimmung der Wahrscheinlichkeitsdichte der Verteilungen wird die Funktion `pdfgmm` aus der STPR-Toolbox verwendet. Sie erwartet als Eingabe die Punkte, an denen die Wahrscheinlichkeitsdichte bestimmt werden soll, als Spaltenvektoren einer Matrix und das mit `mlcgmm` erstellte Modell. Sie liefert die Wahrscheinlichkeitsdichte für diese Punkte in einem Zeilenvektor zurück. Je nachdem bei welcher Verteilung die Wahrscheinlichkeitsdichte größer ist, wird der betrachtete Punkt (Bildausschnitt) der Klasse

Trichter oder Hintergrund zugeordnet.

Handelt es sich bei den klassifizierten Daten um Trainingsdaten, ist die wahre Klassenzugehörigkeit also bekannt, können nun zwei Fehlerraten bestimmt werden.  $\alpha$  entspricht Hintergrundausschnitten, die als Trichter klassifiziert wurden,  $\beta$  entspricht den als Hintergrund klassifizierten Trichtern.  $\alpha$  und  $\beta$  können jeweils absolut und relativ angegeben werden.

Die Klassifizierung und Fehlerratenbestimmung erfolgte auch für die 3600 Trainingsdaten, die für die Erstellung der PCA- und LDA-Modelle und Verteilungsparameter verwendet wurden. Laut Tabelle 3.3 ist also eine Fehlklassifizierung bei den Trichterausschnitten von mehr als 13.3% und eine Fehlklassifizierung der Hintergrundausschnitte von mehr als 10.1% zu erwarten. Die weiteren Angaben zu den Trainingsdaten in Tabelle 3.3 werden im folgenden Abschnitt erläutert.

Faktor	1.5
Fenstergröße	25
# Trichter	900
# Hintergrund	2701
PCA Dim.	45
$\beta$ abs.	120
$\alpha$ abs.	273
$\beta$ rel.	13.3
$\alpha$ rel.	10.1

Tabelle 3.3: Fehlklassifikationsraten bei den 3600 Trainingsmustern

### 3.5 Ausschnittsgröße bei der Klassifikation

Das Template zur Kandidatensuche wurde aus Trichterausschnitten erstellt, die den Rand des Trichters jeweils genau umschließen. Bei der Klassifikation bietet es sich aber an, Ausschnitte zu verwenden, die auch die nähere Umgebung des Kandidaten enthalten. Wir erwarten ein besseres Klassifikationsergebnis dadurch, dass viele Trichter einen charakteristischen Erdauswurf aufweisen, wohingegen die Umgebung der Hintergrundfenster, die mit dem Template als Trichter erkannt wurden, nicht „trichtertypisch“ ist und stark variiert.

Um dies zu verifizieren wird das Training drei Mal mit denselben Trainingsmustern durchgeführt. Beim ersten Mal entspricht die Größe, mit der die Trainingsmuster aus dem Bild ausgeschnitten werden, der Größe des Templates, mit dem sie gefunden wurden. Es wird also keine Umgebung berücksichtigt. Beim zweiten Mal wird die Größe des Templates, mit dem sie gefunden wurden, mit dem Faktor 1.5 multipliziert und dann die Trainingsmuster ausgeschnitten. Damit umschließen die Ausschnitte die Trichter nicht genau, sondern enthalten noch Teile der nahen Umgebung. Beim dritten Mal wird die Größe vor dem Ausschneiden mit dem Faktor 2.0 multipliziert. Um jeweils wieder möglichst wenige Ausschnitte skalieren zu müssen und möglichst die gleiche Anzahl zu vergrößern wie zu verkleinern, werden die Ausschnittsgrößen, auf die skaliert wird, angepasst. Der Faktor 1.0 entspricht dann der Skalierung auf Ausschnittsgröße 17, der Faktor 1.5 der Skalierung auf Ausschnittsgröße 25 und der Faktor 2.0 der Skalierung auf Ausschnittsgröße 33.

Nach dem Training werden die Trainingsdaten klassifiziert und die Fehlerraten bestimmt. Dies wurde für alle drei Faktoren insgesamt sechs Mal durchgeführt, jeweils mit einer anderen Anzahl und/oder Auswahl an Trichter- und Hintergrundausschnitten. Die Ergebnisse sind in Tabelle 3.4 dargestellt. Aufgelistet ist zur Information auch die Dimension der in den PCA-Raum projizierten Ausschnitte. Die Tabelle 3.5 zeigt die mittleren relativen Fehlerraten bei den drei Faktoren. Während die Fehlklassifikation bei Faktor 1.0 erwartungsgemäß höher ist, als bei einem größeren Faktor, unterscheiden sich die Werte der Fehlklassifikation bei den Faktoren 1.5 und 2.0 nicht wesentlich. Es wurde daher aus Laufzeitgründen die Ausschnittsgröße 25 gewählt. Je kleiner die Dimension  $D$  ist, desto kürzer ist die Laufzeit der PCA.

Faktor	1.0	1.5	2.0	1.0	1.5	2.0	1.0	1.5	2.0
Fenstergröße	17	25	33	17	25	33	17	25	33
# Trichter	299	299	299	151	151	151	99	99	99
# Hintergrund	899	899	899	449	449	449	299	299	299
PCA Dim.	23	43	60	21	40	54	20	35	47
$\beta$ abs.	42	39	40	25	18	17	15	10	10
$\alpha$ abs.	113	88	82	51	40	40	31	25	29
$\beta$ rel.	14.0	13.0	13.4	16.6	11.9	11.3	15.2	10.1	10.1
$\alpha$ rel.	12.6	9.8	9.1	11.4	8.9	8.9	10.4	8.4	9.7

Faktor	1.0	1.5	2.0	1.0	1.5	2.0	1.0	1.5	2.0
Fenstergröße	17	25	33	17	25	33	17	25	33
# Trichter	199	199	199	199	199	199	199	199	199
# Hintergrund	598	598	598	598	598	598	598	598	598
PCA Dim.	22	41	55	21	41	55	23	43	59
$\beta$ abs.	31	21	19	30	19	20	28	19	19
$\alpha$ abs.	77	58	69	74	64	57	76	65	64
$\beta$ rel.	15.6	10.6	9.5	15.1	9.5	10.1	14.1	9.5	9.5
$\alpha$ rel.	12.9	9.7	11.5	12.4	10.7	9.5	12.7	10.9	10.7

Tabelle 3.4: Fehlklassifikationsraten bei unterschiedlich großer Umgebung auf den Ausschnitten

Faktor	1.0	1.5	2.0
$\beta$ rel.	15.1	10.8	10.6
$\alpha$ rel.	12.0	9.7	9.9

Tabelle 3.5: Mittlere Fehlklassifikationsraten bei unterschiedlich großer Umgebung auf den Ausschnitten

## 4. Ergebnisse

### Analyse und Bewertung des Algorithmus

In diesem Kapitel soll darauf eingegangen werden, welche Ergebnisse mit dem erarbeiteten Algorithmus zu erzielen waren. Im ersten Teil werden die Resultate der Anwendung des Algorithmus auf die Trainingsbilder dargestellt, um einen Überblick über die Leistungsfähigkeit des Algorithmus zu erhalten. Im zweiten Teil analysieren wir die Ergebnisse bei Anwendung des Algorithmus auf die 30 Testbildern der Testmenge. Dabei wird die Fehlklassifikation genauer untersucht und die Einsatzmöglichkeit des Algorithmus zur Erstellung einer „Bombenrichter-Dichte-Karte“ getestet. Im Folgenden werden zunächst kurz Begriffe und Schemata erläutert, die zur Darstellung der Ergebnisse verwendet werden.

#### 4.1 Allgemeines zur Bewertung

Um die Güte der Klassifikation zu beurteilen, werden die vom Algorithmus detektierten Trichter mit den annotierten Trichtern in den Trainingsbildern verglichen. Bei der Darstellung der Ergebnisse werden Konfusionstabellen verwendet. Die Ergebnisse der Kandidatensuche und die Ergebnisse der Klassifizierung werden getrennt untersucht und dabei verschiedene Konfusionstabellen genutzt.

In der Konfusionstabelle für die Kandidatensuche werden die Ergebnisse der in Abschnitt 3.3.1 vorgenommenen Einteilung der annotierten Trichter und der gefundenen Kandidaten in die drei Kategorien dargestellt (Tabelle 4.1). Bei der Anzahl der Ausschnitte der Kategorie 3 handelt es sich um Gesamtzahl der Hintergrundausschnitte, die später klassifiziert werden. Deshalb ist sie in der rechten unteren Ecke der Konfusionstabelle als wahre Summe der Hintergrundausschnitte angegeben. Die Einteilung in Trichter und Hintergrund wird erst bei der Klassifizierung vorgenommen, deshalb sind die beiden Einträge darüber noch nicht gefüllt.

		Referenzdaten	
		T	H
Klassifizierte Daten	T	Kat. 1	-
	H	Kat. 2	-
$\Sigma$		Kat. 3	

Tabelle 4.1: Beispiel für eine Konfusionstabelle bei der Kandidatensuche

Bei der Klassifizierung werden alle Kandidaten in die Klassen Trichter und Hintergrund eingeteilt. Die Konfusionstabelle kann beim Vergleich mit dem Trainingsbild also vollständig gefüllt werden. Ein Beispiel zeigt die Tabelle 4.2. Der Einfachheit halber bezeichnen wir im Folgenden die Bildausschnitte,

- die als Trichter klassifiziert wurden und die auch Trichter darstellen mit *right positives*. Ihre Anzahl wird in Tabelle 4.2 mit a angegeben.
- die als Trichter klassifiziert wurden, obwohl sie Hintergrund darstellen, mit *false positives*. Ihre Anzahl wird in Tabelle 4.2 mit c angegeben.
- die als Hintergrund klassifiziert wurden und die auch Hintergrund darstellen mit *right negatives*. Ihre Anzahl wird in Tabelle 4.2 mit d angegeben.
- die als Hintergrund klassifiziert wurden, obwohl sie Trichter darstellen, mit *false negatives*. Ihre Anzahl wird in Tabelle 4.2 mit b angegeben.

		Referenzdaten		
		T	H	$\Sigma$
Klassifizierte Daten	T	a	c	a+c
	H	b	d	b+d
$\Sigma$		a+b	c+d	a+b+c+d

Tabelle 4.2: Beispiel für eine Konfusionstabelle bei der Klassifikation mit absoluten Werten

Neben der Konfusionstabelle mit den absoluten Klassifikationsergebnissen wird auch eine für die relativen Werte verwendet (Tabelle 4.3). Der Anteil der false positives an der Gesamtzahl der Hintergrundausschnitte ist die Fehlerrate  $\alpha$ , oder auch Fehler 1. Art. Sie berechnet sich also nach  $\frac{c}{c+d}$ . Der Anteil der false negatives an der Gesamtzahl der Trichterausschnitte ist die Fehlerrate  $\beta$ , oder auch Fehler 2. Art und berechnet sich nach  $\frac{b}{a+b}$ . Eine wichtige Angabe, die in der relativen Konfusionstabelle abgelesen werden kann, ist die Relevanz. Sie berechnet sich nach  $\frac{a}{a+c}$  und gibt die Wahrscheinlichkeit an, dass ein Bildausschnitt, der als Trichter erkannt wurde, auch wirklich ein Trichter ist. In Tabelle 4.3 wird im Eintrag unter der Relevanz auch die Segreganz angegeben, die die Wahrscheinlichkeit angibt, dass ein Bildausschnitt, der bei der Klassifikation als Hintergrund eingeordnet wurde, auch wirklich kein Trichter ist. Die Segreganz berechnet sich nach  $\frac{d}{b+d}$ .

		Referenzdaten		
		T [%]	H [%]	Rel. [%]
Klassifizierte Daten	T [%]	a/(a+b)	c/(c+d)	a/(a+c)
	H [%]	b/(a+b)	d/(c+d)	d/(b+d)
$\Sigma$ [%]		100	100	

Tabelle 4.3: Beispiel für eine Konfusionstabelle bei der Klassifikation mit relativen Werten

## 4.2 Ergebnisse bei der Kandidatensuche

In diesem Abschnitt soll dargestellt werden, welche Ergebnisse bei der Kandidatensuche erzielt wurden. Die Kandidatensuche mittels Kreuzkorrelation stellt die Vorauswahl der Bildausschnitte dar. Optimalerweise sollte sie also möglichst alle Bombentrichter erfassen.

### 4.2.1 Verschiedene Templateanzahlen

Die Kandidatensuche mit den 5 Templategrößen 33, 23, 17, 11 und 9 liefert sehr viele Kandidaten, von denen die meisten Hintergrund sind. Die Tabelle 4.4 zeigt für jedes Bild der Testmenge die Anzahl der Ausschnitte in den drei Kategorien (vgl. Abschnitt 3.3.1). Die Summe wurde dabei einmal über alle Bilder der Gruppe 1 (vgl. Abschnitt 3.1.4) und einmal über alle Bilder der Gruppe 2 gebildet. Die hohe Anzahl in der Kategorie 3 ist problematisch, da sie bei der Klassifikation dazu führt, dass trotz eines geringen relativen Fehlers  $\alpha$  die Absolutzahl der false positives im Vergleich zu den right positives sehr hoch ist. Die Relevanz der Detektion ist dadurch sehr gering. Bei der Analyse der Kandidatensuche zeigte sich, dass mit der Templategröße 9 besonders viele Kandidaten gefunden werden, die fast alle in die Kategorie 3 eingeordnet werden. Deshalb wurden die mit der Templategröße 9 gefundenen Kandidaten ausgeschlossen und die Einteilung in die drei Kategorien erneut durchgeführt. Das Ergebnis zeigt ebenfalls die Tabelle 4.4.

Die Summen über die Kategorien sind für die Gruppe 1 und die Gruppe 2 in der Konfusionsstabelle 4.5 dargestellt. Es wurde jeweils die Fehlerrate  $\beta$  in Prozent berechnet, die die Prozentzahl der bei der Kandidatensuche nicht erfassten Trichter angibt. Sie liegt, wenn alle 5 Templategrößen verwendet werden, bei 19.5%, beim Weglassen der Templategröße 9 bei ca. 24.2%. Der Wert der Gruppe 2 wird als aussagekräftiger beurteilt, da diese Gruppe mehr Bilder enthält. Mit dem verwendeten Template können also nur etwa 80.5% bzw. 75.8% der Trichter erfasst werden. Es ist nicht erstaunlich, dass mit dem Template nicht alle Trichter gefunden werden, da sie eine sehr unterschiedliche Gestalt haben (Abschnitt 1.2.1). Das Konzept wurde aber so erstellt, dass es leicht auf mehrere Trichterklassen angepasst werden kann. Die Suche würde dann mit mehreren Templates durchgeführt, und die Detektionsrate könnte gesteigert werden.

Es wurde außerdem das Verhältnis der Kategorie 1 zur Kategorie 3 bestimmt, das wir mit  $\nu$  bezeichnen.  $\nu$  gibt einen Überblick über das Verhältnis von Trichtern zu Hintergrund bei den Kandidaten, was beim Training berücksichtigt werden kann (Abschnitt 3.3.2). Es ist auch ein Maß für die Relevanz, die bei der Klassifikation erreicht werden kann. Je größer  $\nu$  ist, desto weniger Hintergrund wird bei der Kandidatensuche mit dem Template erfasst und desto größer wird die Relevanz.

$\nu$  ist hier in allen Fällen sehr klein (zwischen 0.019 und 0.101), das Verhältnis von Trichtern zu Hintergrund liegt bei den Kandidaten also zwischen 1:53 und 1:10. Durch Weglassen der Templategröße 9 konnte  $\nu$  immerhin jeweils verdoppelt werden. Auffällig ist, dass  $\nu$  bei der Gruppe 2 größer ist, obwohl sich in dieser Gruppe auch Bilder befinden, die nicht vollständig annotiert sind und es somit möglich ist, dass sich in der Kategorie 3 noch Trichterausschnitte befinden. Man würde erwarten, dass die Anzahl der Ausschnitte in Kategorie 3 deshalb noch größer ist. Dass  $\nu$  trotzdem größer ist, liegt daran, dass auf den Bildern der Gruppe 1 insgesamt wenige Trichter sind und auf den Bildern, die nicht vollständig annotiert wurden, viele Trichter



	Alle 5 Templategrößen			Ohne Templategröße 9		
Bildnr.	Kat. 1	Kat. 2	Kat. 3	Kat. 1	Kat. 2	Kat. 3
255515	0	0	526	0	0	258
261349	7	0	523	7	0	213
266763	0	0	387	0	0	153
266882	117	26	339	115	28	125
266892	0	0	422	0	0	155
267014	17	2	575	17	2	241
267020	8	4	419	8	4	184
268510	16	2	889	13	5	426
268536	12	1	776	11	2	336
268847	0	0	1190	0	0	582
268853	8	3	1246	8	3	533
268857	15	1	1152	15	1	537
268859	21	5	1305	19	7	625
271615	0	0	1002	0	0	405
274840	0	0	2060	0	0	1037
277329	35	17	444	35	17	186
$\Sigma$ Gr. 1	256	61	13255	248	69	5996
259795	131	24	1184	127	28	582
259865	230	52	1632	230	52	902
266791	51	10	804	48	13	364
266797	111	15	778	108	18	411
266820	25	11	647	24	12	274
266877	100	38	422	99	39	192
268861	69	12	1662	54	27	859
277412	84	33	581	57	60	272
$\Sigma$ Gr. 2	1057	256	20965	995	318	9852

Tabelle 4.4: Ergebnis der Kandidatensuche (mit verschiedener Templatezahl) nach Vergleich mit dem Trainingsbild

Alle 5 Templategrößen			Ohne Templategröße 9		
Gr. 1	T	H	Gr. 1	T	H
T	256	-	T	248	-
H	61	-	H	69	-
$\Sigma$	317	13255	$\Sigma$	317	5996
Gr. 2	T	H	Gr. 2	T	H
T	1057	-	T	995	-
H	256	-	H	318	-
$\Sigma$	1313	20965	$\Sigma$	1313	9852
	$\beta[\%]$	$\nu$		$\beta[\%]$	$\nu$
Gr. 1	19.24	0.019	Gr. 1	21.77	0.041
Gr. 2	19.50	0.050	Gr. 2	24.22	0.101

Tabelle 4.5: Konfusionstabellen der Kandidatensuche für beide Gruppen bei verschiedener Templatezahl

markiert sind. Während die mittlere Trichteranzahl in Gruppe 2 also (bei relativer Betrachtung) wesentlich größer ist, als in Gruppe 1, ist die mittlere Anzahl der Hintergrundausschnitte bei Gruppe 2 nur geringfügig höher als bei Gruppe 1<sup>1</sup>.

Bei der Kreuzkorrelation wurde der Schwellwert  $\tau$  auf 0.7 gesetzt. Bei weiteren Untersuchungen könnte festgestellt werden, wie sich die Detektionsrate verschlechtert und  $\nu$  gesteigert wird, wenn  $\tau$  auf einen größeren Wert gesetzt wird. Bei der Verwendung von mehreren Templates ist zu vermuten, dass  $\tau$  ohne starken Anstieg der Fehlerrate  $\beta$  erhöht werden kann und  $\nu$  dabei größer wird.

#### 4.2.2 Rotationsinvarianz des Templates

Aufgrund der unterschiedlichen Sonnenstände, Flug- und Aufnahmerrichtungen bei den Bildflügen wurden Bombentrichter aus verschiedenen Perspektiven aufgenommen. Insbesondere der sichel- oder ellipsenförmige Schattenwurf ist jeweils anders ausgerichtet. Es ist naheliegend, die vorhandenen Bildausschnitte der annotierten Trichter jeweils um  $90^\circ$ ,  $180^\circ$  und  $270^\circ$  zu drehen um somit zusätzliche Beispiele für Trichter zu erhalten, die scheinbar aus einer anderen Perspektive aufgenommen wurden. Dadurch kann das Template aus der vierfachen Anzahl an Trichterausschnitten erstellt werden und wird damit rotationsinvariant bezüglich der Drehung um Vielfache von  $90^\circ$ . Abbildung 4.1 zeigt links das rotationsinvariante Template und rechts das Template, das aus den Trichterausschnitten in ihrer ursprünglichen Orientierung erstellt wurde.

Die Kandidatensuche wurde mit beiden Templates durchgeführt und die gefundenen Kandidaten (ohne Templategröße 9) mit den annotierten Trainingsbildern verglichen. Die Ergebnisse sind in Tabelle 4.6 dargestellt. Danach ergibt sich insgesamt eine Detektionsrate von 74.0% für das rotationsinvariante Template und eine Detektionsrate von 75.8% für das nicht rotationsinvariante Template.

Rotationsinv. Template			Nicht rotationsinv. Template		
Gr. 2	T	H	Gr. 2	T	H
T	972	-	T	995	-
H	341	-	H	318	-
$\Sigma$	1313	9372	S	1313	9852
$\beta$ [%]			$\beta$ [%]		
$\nu$			$\nu$		
25.97			24.22		
0.104			0.101		

Tabelle 4.6: Konfusionstabellen der Kandidatensuche bei rotationsinvariantem und nicht rotationsinvariantem Template

Der Grund für die höhere Detektionsrate mit dem nicht rotationsinvarianten Template könnte darin liegen, dass die meisten Trichter nicht rotationsinvariant sind. Es gibt hingegen viele, deren Inneres aufgrund des Schattenwurfs eher der leicht elliptischen Form des nicht rotationsinvarianten Templates entspricht. Für die weiteren Berechnungen wurde hier aufgrund der geringfügig besseren Detektionsrate das nicht rotationsinvariante Template verwendet, das in Abbildung 4.1 rechts dargestellt ist. Später sollen mehrere Templates verwendet werden, so dass dann z.B.

<sup>1</sup>Bei allen fünf Templategrößen werden im Mittel 828 Hintergrundausschnitte in Gruppe 1 und 873 in Gruppe 2 gefunden. Die Steigerung liegt bei 5.4%. Die Zahl der gefundenen Trichter liegt in Gruppe 1 im Mittel bei 11, in Gruppe 2 bei 44. Damit tritt eine Steigerung um 175% auf.

auch die unterschiedliche Lage des Schattens durch verschiedene Templates berücksichtigt werden kann und sich die Frage nach Rotationsinvarianz erübrigt. Hier ist zu erwähnen, dass bei mehreren Trichterklassen trotzdem die Drehung der Trichterausschnitte sinnvoll sein kann, um mehr Beispiele einer Klasse zu erhalten. Die vier Ausschnitte, die aus einem Trichter erhalten werden, würden in diesem Fall aber u.U. vier verschiedenen Klassen zugeordnet und führten nicht zur Rotationsinvarianz der Templates.

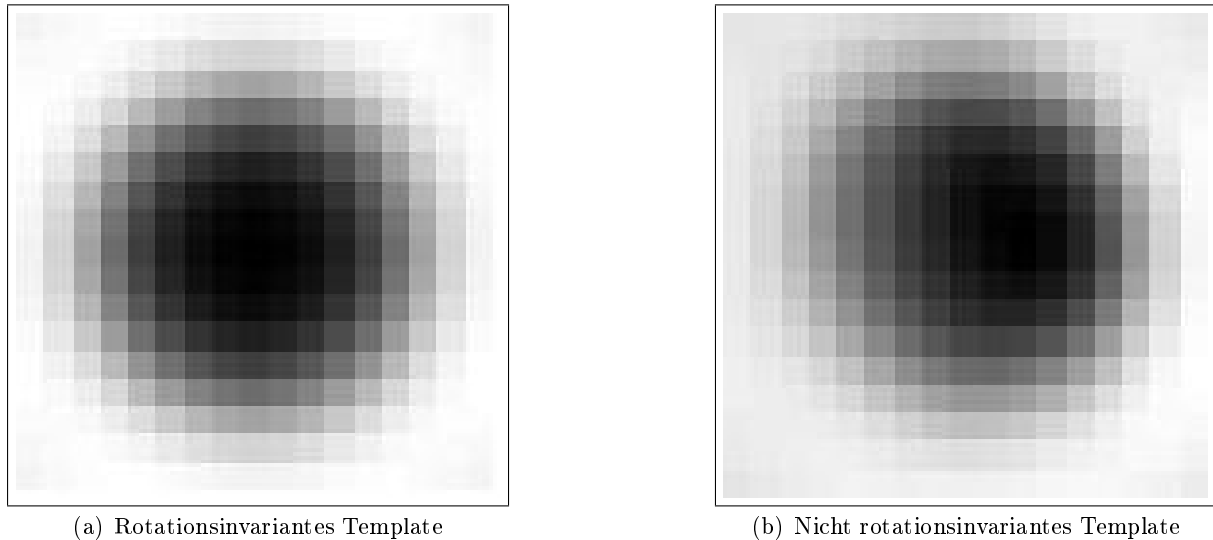


Abbildung 4.1: Mittlerer Trichter der Größe  $17 \times 17$  Pixel

### 4.3 Ergebnisse der Klassifikation

Die Klassifizierung wurde auf den Trainingsbildern zweimal durchgeführt: einmal mit den Kandidaten, die mit allen fünf Templategrößen gefunden wurden und einmal mit den Kandidaten ohne die mit der Templategröße 9 gefundenen. Die Ergebnisse wurden mit den Annotationen verglichen und somit die Anzahlen a, b, c und d der right positives, false negatives, false positives und right negatives festgestellt. Sie sind für jedes Bild in Tabelle 4.7 aufgelistet. Danach wurden die absoluten und relativen Konfusionstabellen jeweils für die Gruppe 1 und die Gruppe 2 erstellt, einmal für die Suche mit allen Templates und einmal für die Suche ohne Templategröße 9 (Tabellen 4.8 und 4.9).

Für die Beurteilung der right positives und false negatives (a und b), sowie der Fehlerrate  $\beta$  wird die Gruppe 2 mit allen Trainingsbildern verwendet, weil sie die größere Stichprobe darstellt. Bei der Gruppe 2 befinden sich in der Kategorie 3 aufgrund der unvollständigen Annotation höchstwahrscheinlich nicht nur Hintergrund- sondern auch Trichterausschnitte. Nach der Aufteilung der Ausschnitte der Kategorie 3 in die false positives und right negatives enthalten diese Mengen jeweils noch right positives bzw. false negatives. Es kommt dadurch zur Verfälschung der Fehlerrate  $\alpha$  und der Relevanz. Deshalb werden diese Werte überwiegend anhand der Ergebnisse aus Gruppe 1 beurteilt.

Es wird erwartet, dass sich die Relevanz verschlechtert, wenn sich unter den false positives noch right positives befinden. Dies ist allerdings aufgrund der verschiedenen mittleren Trichteranzahlen

(a) alle 5 Templategrößen					(b) ohne Templategröße 9				
Bildnr.	a	b	c	d	Bildnr.	a	b	c	d
255515	0	0	50	476	255515	0	0	30	228
261349	7	0	67	456	261349	7	0	47	166
266763	0	0	32	355	266763	0	0	19	134
266882	103	40	89	250	266882	98	45	40	85
266892	0	0	36	386	266892	0	0	23	132
267014	11	8	96	479	267014	9	10	44	197
267020	7	5	48	371	267020	7	5	19	165
268510	15	3	74	815	268510	13	5	51	375
268536	10	3	88	688	268536	10	3	44	292
268847	0	0	65	1125	268847	0	0	33	549
268853	8	3	93	1153	268853	8	3	48	485
268857	14	2	86	1066	268857	14	2	45	492
268859	20	6	148	1157	268859	18	8	84	541
271615	0	0	106	896	271615	0	0	48	357
274840	0	0	126	1934	274840	0	0	66	971
277329	28	24	42	402	277329	28	24	20	166
$\Sigma$ Gr. 1	223	94	1246	12009	$\Sigma$ Gr. 1	212	105	661	5335
259795	122	33	116	1068	259795	120	35	90	492
259865	187	95	567	1065	259865	181	101	382	520
266791	45	16	159	645	266791	43	18	88	276
266797	103	23	270	508	266797	86	40	154	257
266820	20	16	89	558	266820	20	16	43	231
266877	88	50	109	313	266877	86	52	59	133
268861	61	20	371	1291	268861	47	34	233	626
277412	69	48	341	240	277412	41	76	158	114
$\Sigma$ Gr. 2	918	395	3268	17697	$\Sigma$ Gr. 2	836	477	1868	7984

Tabelle 4.7: Absolute Klassifikationsergebnisse bei allen Trainingsbildern mit verschiedener Templatezahl

auf den Bildern in Gruppe 1 und Gruppe 2 nicht der Fall (vgl. Abschnitt 4.2.1). Damit ist die Relevanz also abhängig von der wahren Anzahl der Trichter im Bild. Wird ein Ausschnitt aus einem Bild mit vielen Trichtern als Trichter klassifiziert, ist die Wahrscheinlichkeit höher, dass es sich wirklich um einen Trichter handelt.

Selbst beim Ausschluss von Templategröße 9 und bei der Gruppe 2 mit vielen Trichtern liegt die Relevanz nur bei knapp 31%. Es werden bei dieser Konfiguration insgesamt knapp 64% aller Trichter mit dem Algorithmus detektiert, in 69% aller Fälle ist ein detektierter Trichter aber kein Trichter. An den Konfusionstabellen kann man ablesen, dass Relevanz und Detektionsrate sich antiproportional zueinander verhalten. Bei der Suche mit allen Templates in Gruppe 1 wurden zwar 70% der Trichter gefunden, allerdings nur mit einer Relevanz von 15%.

Die Fehlerrate  $\beta$ , die angibt, wieviel Prozent der Trichter nicht gefunden werden, liegt für die Gruppe 2 bei 30.1% bzw. 36.3%. Zieht man von diesen Werten jeweils die Fehlerrate  $\beta$ , die aus der Kandidatensuche resultiert, ab, so erhält man Werte von 10.6% und 12.1%. Erwartet war eine Fehlerrate  $\beta$  der Klassifizierung von 13.3% (vgl. Tabelle 3.3). Wahrscheinlich handelt es sich um zufällige Abweichungen.

Die Segreganz liegt für die verschiedenen Konfigurationen zwischen 94% und 99%. Ein bei

der Klassifikation ausgeschlossener Kandidat ist also in den allermeisten Fällen auch wirklich Hintergrund. Die Fehlerrate  $\alpha$  liegt für die Gruppe 1 bei 9.4% bzw. 11.0%. Dies entspricht der bei der Klassifizierung zu erwartenden Fehlerrate von 10.1% (vgl. Tabelle 3.3).

Bei der Weiterentwicklung des Algorithmus muss die Relevanz gesteigert werden, um eine aussagekräftige Trichterdetektion zu erhalten<sup>2</sup>. Im folgenden Abschnitt werden die false positives näher untersucht, um einen Überblick über die Bildstrukturen zu erhalten, die für die geringe Relevanz sorgen.

(a) absolut				(b) relativ			
Gr. 1	T	H	$\Sigma$	Gr. 1	T [%]	H [%]	Rel. [%]
T	223	1246	1469	T [%]	70.3	9.4	15.2
H	94	12009	12103	H [%]	29.7	90.6	99.2
$\Sigma$	317	13255	13572	$\Sigma$ [%]	100.0	100.0	
Gr. 2	T	H	$\Sigma$	Gr. 2	T [%]	H [%]	Rel. [%]
T	918	3268	4186	T [%]	69.9	15.6	21.9
H	395	17697	18092	H [%]	30.1	84.4	97.8
$\Sigma$	1313	20965	22278	$\Sigma$ [%]	100.0	100.0	

Tabelle 4.8: Konfusionstabellen für die Klassifikation bei Suche mit allen Templates

(a) absolut				(b) relativ			
Gr. 1	T	H	$\Sigma$	Gr. 1	T [%]	H [%]	Rel. [%]
T	212	661	873	T [%]	66.9	11.0	24.3
H	105	5335	5440	H [%]	33.1	89.0	98.1
$\Sigma$	317	5996	6313	$\Sigma$ [%]	100.0	100.0	
Gr. 2	T	H	$\Sigma$	Gr. 2	T [%]	H [%]	Rel. [%]
T	836	1868	2704	T [%]	63.7	19.0	30.9
H	477	7984	8461	H [%]	36.3	81.0	94.4
$\Sigma$	1313	9852	11165	$\Sigma$ [%]	100.0	100.0	

Tabelle 4.9: Konfusionstabellen für die Klassifikation bei Suche ohne Templategröße 9

## 4.4 Analyse der Fehlklassifikation

Die Bombenrichterdetektion wurde auf 30 Testbildern durchgeführt. Dabei wurden das Template, die Projektionsmatrizen, sowie die Parameter der Verteilungen verwendet, die aus den Trainingsbildern ermittelt wurden. Die gefundenen Trichter wurden bei der Anzeige der Bilder jeweils mit einem roten Kästchen markiert. So ist die optische Analyse der Fehlklassifikation möglich. Dabei wurde festgestellt, dass sowohl bei den false positives, als auch bei den false negatives bestimmte Strukturen häufig auftreten. In den folgenden beiden Abschnitten werden sie kategorisiert und Beispiele anhand von Bildausschnitten gezeigt. Mehr Beispiele für false positives und false negatives befinden sich im Anhang.

<sup>2</sup>Für die Erstellung einer „Bombenrichter-Dichte-Karte“ müsste dies nicht geschehen, wenn das Verhältnis  $\nu$  konstant wäre. Dies ist aber bei den verwendeten Trainingsbildern nicht der Fall.

#### 4.4.1 False positives

Dieser Abschnitt gibt Beispiele für Hintergrundausschnitte, die vom Algorithmus als Trichter erkannt wurden. Die Grauwerte in diesen Bildausschnitten erfüllen die Kriterien, dass sie sowohl mit dem Template als Kandidaten gefunden werden, als auch bei der Klassifikation nicht in die Klasse Hintergrund eingeordnet werden. Der Mensch hat allerdings den Vorteil, dass er die fragliche Region in einem größeren Kontext betrachten, und sie z.B. auch mit dem Aussehen benachbarter Trichter oder anderer Objekte vergleichen kann. So kann er viele Stellen, die von der Grauwertverteilung her Trichter sein könnten, aufgrund des Kontextes verwerfen. Der Algorithmus stützt sich allein auf die Grauwerte im Bildfenster. Bei der Analyse der 30 Testbilder lassen sich Objekte in bestimmten Kontexten identifizieren, die immer wieder zu false positives führen.

##### 4.4.1.1 Schattenwurf an Straßen

False positives entstehen oft durch Schattenwurf von Gebäuden, Sträuchern oder Bäumen, die sich an Straßen und Wegen befinden (Abbildung 4.2). Der Grund dafür ist, dass Straßen auf den Bildern hell sind und sich somit der ellipsenartige Schattenwurf auf einem weißen Hintergrund befindet.

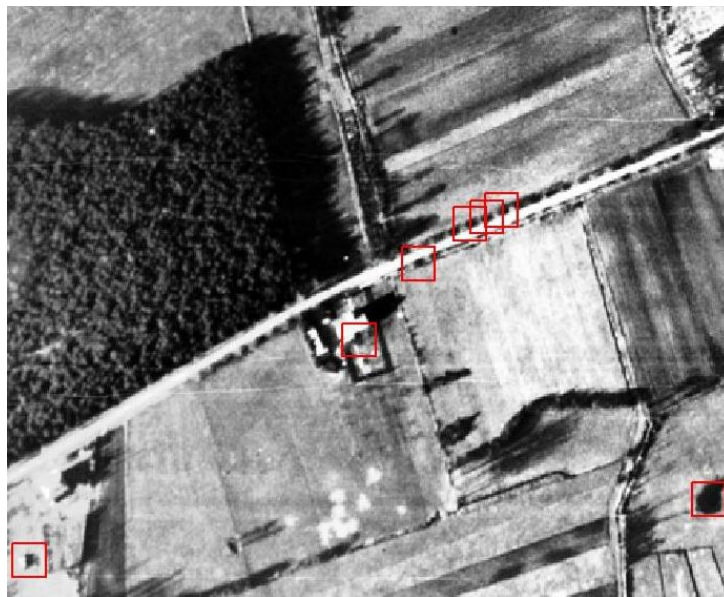


Abbildung 4.2: False positives an einer Strasse (Bild 268843)

##### 4.4.1.2 Hausdächer mit angrenzendem Schattenwurf

Häufig stellen die false positives Satteldächer von Gebäuden mit angrenzendem Schattenwurf dar (Abbildung 4.3). Die Dachseite, die bei der Aufnahme von der Sonne beleuchtet wurde, ist auf den Bildern sehr hell, die Dachseite, die im Schatten liegt, verschmilzt mit dem Schattenwurf des Hauses und bildet einen oft abgerundeten Fleck, der Trichterähnlichkeit hat.



Abbildung 4.3: False positives an Hausdächern (Bild 266783)

#### 4.4.1.3 Textur von Feldern

Mit der größten Templategröße werden teilweise Strukturen in Äckern und Feldern als Trichter erkannt (Abbildung 4.4). Die Strukturen stellen sich im Ganzen betrachtet eher streifenförmig dar, aufgrund von Unregelmäßigkeiten in der Bepflanzung entstehen auf dem Bild aber auch Flecken, die von der Grauwertverteilung her Trichter sein können. Für den Menschen ist die Fehlklassifikation an diesen Stellen sofort ersichtlich.



Abbildung 4.4: False positives auf einem Feld (Bild 268851)

#### 4.4.1.4 Wald

In Waldgebieten befinden sich häufig trichterähnliche Bildstrukturen, die z.B. durch einzeln stehende Bäume zustande kommen (Abbildung 4.5). Für den menschlichen Betrachter ist hier auch oft nur durch den Kontext die Identifizierung als Hintergrund (also Wald) möglich.

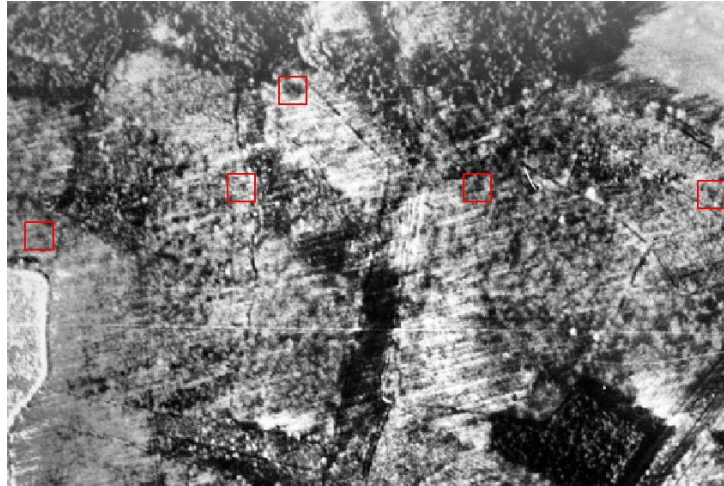


Abbildung 4.5: False positives im Wald (Bild 268855)

#### 4.4.1.5 Nicht eindeutige Bildregionen

Bei Bildausschnitten, die fälschlicherweise als Trichter klassifiziert wurden, handelt es sich oft um Objekte, die auch vom menschlichen Betrachter nicht eindeutig zu identifizieren sind. Bei den Beispielen in Abbildung 4.6 ist nicht auf den ersten Blick klar, ob es sich um einen Trichter handelt oder nicht.



Abbildung 4.6: False positives, die nicht eindeutig zu interpretieren sind (Bild 266781)



#### 4.4.1.6 Rand des Bildes

In nahezu jedem Bild werden am Rand vor allem bei der Beschriftung einige false positives gefunden (Abbildung 4.7). Es handelt sich um weiße Schrift auf schwarzem Hintergrund, des-

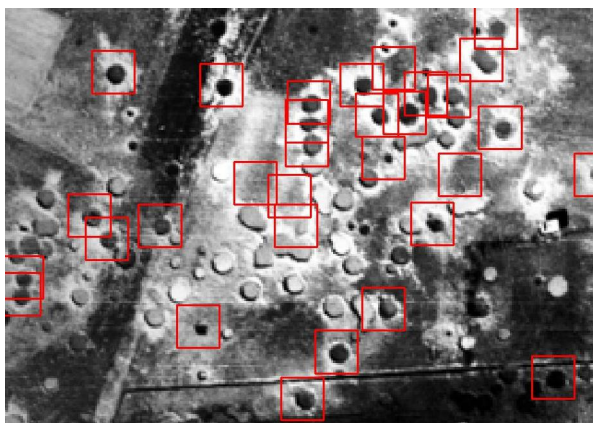


Abbildung 4.7: False positives am Rand (Bild 266879)

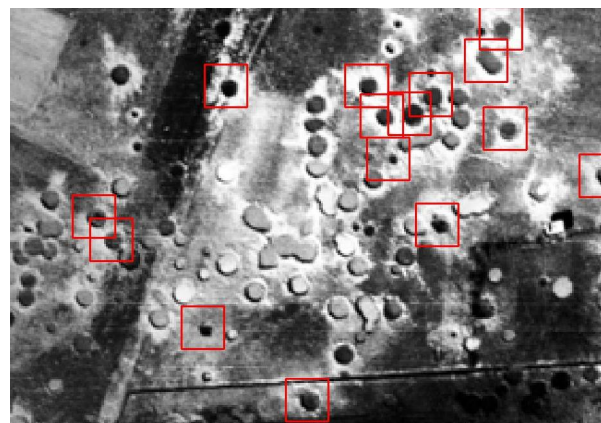
halb ähneln beispielsweise die von den Ziffern 6 oder der 8 eingeschlossenen Hintergrundbereiche schwarzen Kreisscheiben auf weißem Grund und werden als Trichter klassifiziert. Da der Rand nicht bei allen Bildern gleich groß ist, ist es nicht möglich ihn automatisch bei der Suche nach Trichtern auszuschließen. Dafür wäre entweder eine manuelle Vorverarbeitung, oder ein Algorithmus, der den Bildinhalt vom Rand trennen kann, notwendig.

#### 4.4.2 False negatives

In diesem Abschnitt soll ein Überblick gegeben werden, welche Trichter mit dem implementierten Verfahren nicht erkannt werden, und mögliche Ursachen erläutert werden. Außerdem werden Vorschläge für weitere Trichterklassen abgeleitet. Zur Veranschaulichung werden Bildausschnitte gezeigt. Um einen Überblick zu erhalten, welche Trichter bereits bei der Vorauswahl nicht erfasst werden, und welche erst durch Fehlklassifikation ausgeschlossen werden, wird jeweils derselbe Bildausschnitt einmal nach der Kandidatensuche und einmal nach der Klassifikation dargestellt.



(a) nach Kandidatensuche



(b) nach Klassifikation

Abbildung 4.8: „Weiße Bombentrichter“ als false negatives (Bild 259853)

##### 4.4.2.1 Weiße Trichter

Da das Template einer schwarzen Kreisscheibe mit verblassendem Rand auf weißem Grund ähnelt, können Trichter, deren Inneres im Vergleich zum Äußeren nicht dunkel ist, nicht erfasst

werden. Trichter, die mit Wasser gefüllt sind, stellen sich aufgrund der Reflektion des Sonnenlichts oft mit weißem Inneren dar (Abbildung 4.8). Häufig sind die Einschlaglöcher aber auch nicht so tief, deshalb entsteht kein annähernd runder Schattenwurf im Innern, sondern ein schmaler sichelförmiger. Der Rest des Trichters ist dann eher hell, und wird nicht detektiert. Da „weiße“ Trichter unter den false negatives am häufigsten vorkommen, ist es sicher sinnvoll ein Template für diese Klasse zu erstellen.

#### 4.4.2.2 Schwarze Trichter

Viel seltener als „weiße“ Trichter treten als false negatives Trichter auf, die insgesamt zu dunkel sind (Abbildung 4.9). Sie werden aufgrund des fehlenden (oder zu schwachen) weißen Randes bereits vom Template nicht erfasst. Es ist zu überlegen, ob auch für diese Trichter eine neue Klasse eingeführt werden kann. Die Schwierigkeit würde bei der Annotation in der Abgrenzung dieser Trichter von der Klasse der Trichter mit schwarzem Inneren und weißen Rand bestehen.

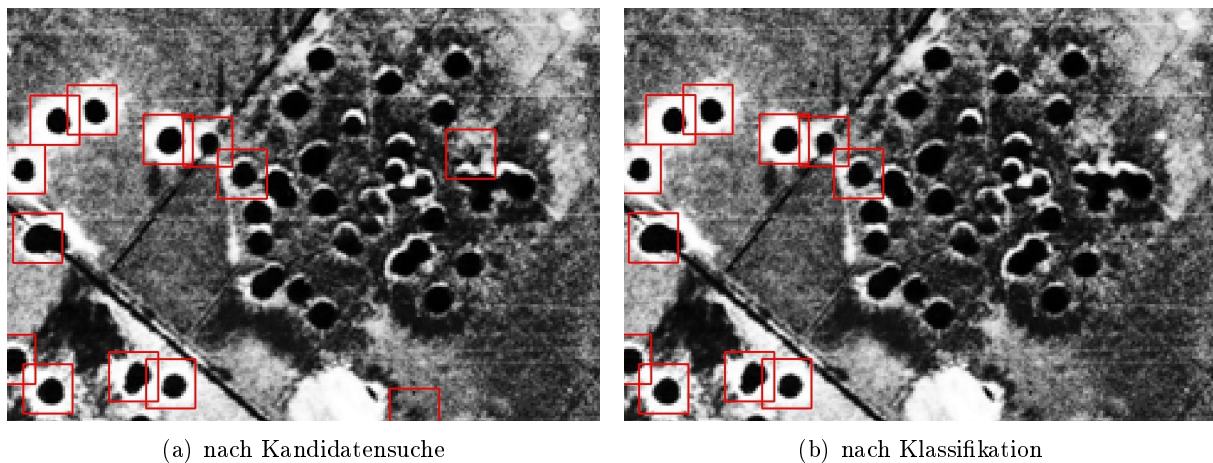


Abbildung 4.9: „Schwarze Bombentrichter“ als false negatives (Bild 255532)

#### 4.4.2.3 Kontrastarme Trichter

Am Rand des Bildes befinden sich oft Trichter, die nicht kontrastreich genug sind, um erkannt zu werden (Abbildung 4.10). Sie werden aber häufig erst bei der Klassifikation ausgeschlossen. Würde man eine Klasse für kontrastarme Trichter vorsehen, könnte das Training mit Beispielen dieser Klasse erfolgen, und sie würden bei der Klassifikation nicht mehr herausfallen.

#### 4.4.2.4 Kleine Trichter

Einige Trichter werden nicht erkannt, weil sie zu klein sind (Abbildung 4.11). Diese false negatives sind schwer auszuschließen, da es sich um Kleinststrukturen im Bild handelt. Diese allein aufgrund ihrer Grauwertverteilung zu detektieren und vor allem von anderen Kleinststrukturen zu unterscheiden, ist extrem aufwendig, und ohne Kontextbetrachtung wahrscheinlich sogar unmöglich. Genau aus diesem Grund wurde der Algorithmus nicht für die Suche nach Blindgängern geschrieben. Die Blindgänger-Einschlaglöcher ähneln den Bombentrümmern der kleinen Bomben und sind klein und unauffällig.

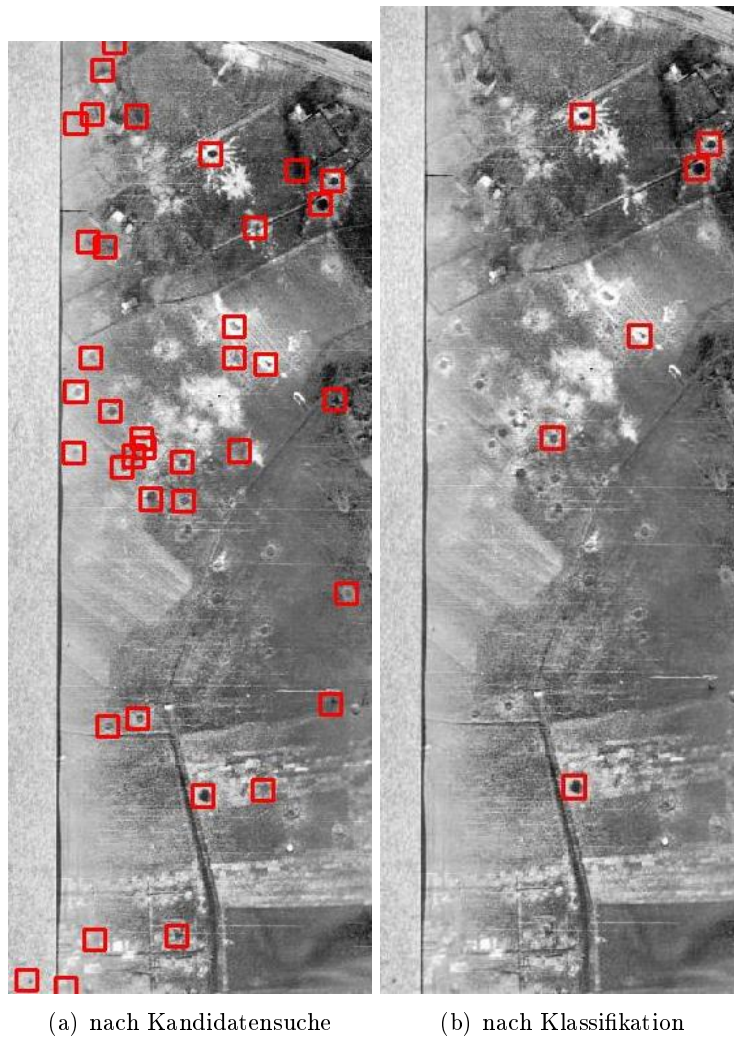


Abbildung 4.10: Kontrastarme Trichter als false negatives (Bild 267008)

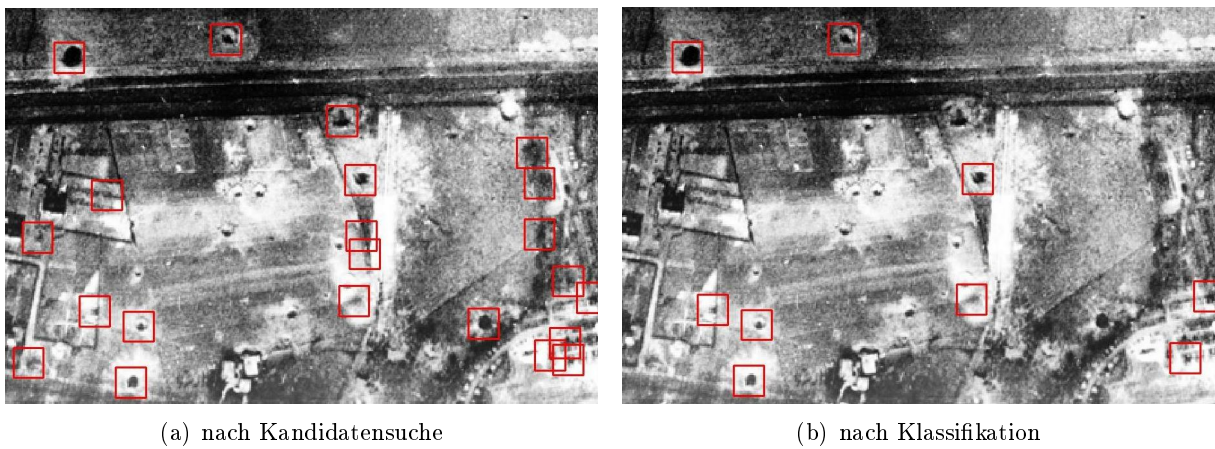


Abbildung 4.11: Kleine Trichter als false negatives (Bild 267008)



#### 4.4.2.5 Uninterpretierbare false negatives

Manchmal liegen fast identisch aussehende Bombentrichter direkt nebeneinander und einige davon werden detektiert, andere nicht (Abbildung 4.12). Diese false negatives sind dann für den menschlichen Betrachter nicht interpretierbar. Oft liegt es nicht daran, dass sie mit dem Template nicht gefunden werden, sondern dass sie bei der Klassifikation aussortiert werden. Würde ein komplexeres Klassifikationsverfahren verwendet, könnte die Fehlklassifikation wahrscheinlich verbessert werden, ohne eine neue Trichterklasse einzuführen.

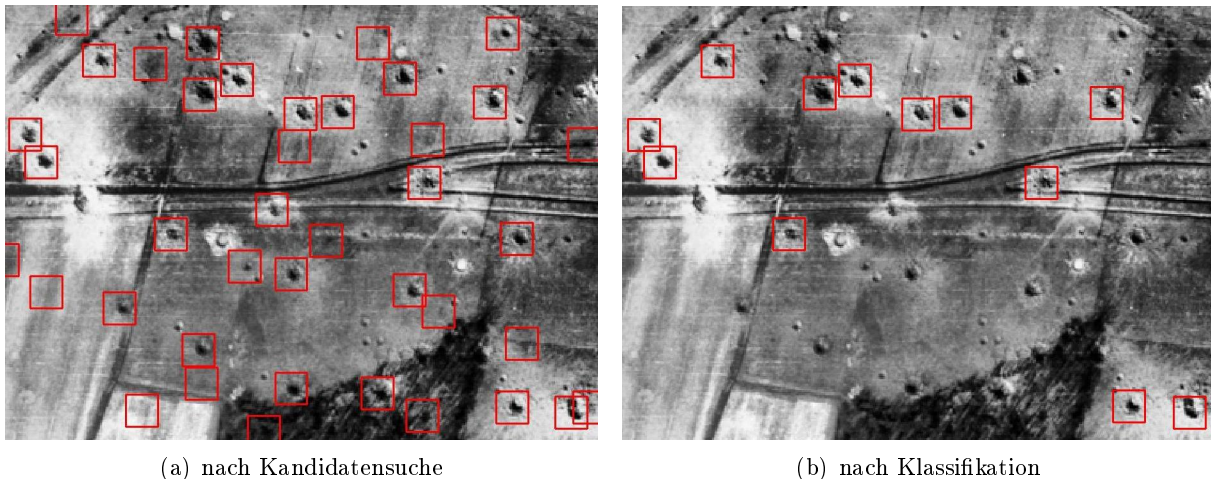


Abbildung 4.12: Uninterpretierbare als false negatives (Bild 259853)

### 4.5 Tests mit neuen Bildern

Um beurteilen zu können, wie gut die Trichterdetektion auf Bildern funktioniert, die nicht in der Trainingsmenge enthalten waren, hätten diese eigentlich vollständig annotiert werden müssen. Dies war aus Zeitgründen bisher nicht möglich. Damit dennoch eine Beurteilung der Ergebnisse der Detektion möglich ist, wurde die Analyse über die „Bombentrichter-Dichte“ vorgenommen.

Da das Ziel des Kampfmittelbeseitigungsdienst die Detektion von Blindgängern ist, ist die genaue Position jedes einzelnen Bombentrichters also von nachrangigem Interesse. Es wäre ausreichend, eine „Bombentrichter-Dichte-Karte“ zu erstellen. Diese soll aussagen, ob ein Gebiet stark, mittel, schwach oder gar nicht bombardiert wurde. Somit wäre es möglich, die manuelle Suche nach Blindgängern auf die relevanten Gebiete einzuschränken.

Um einen ersten Überblick über die Möglichkeit der Einordnung der Bilder nach der Stärke der Bombardierung zu bekommen und damit die Güte der Trichterdetektion auf den Testbildern festzustellen, erfolgte ein Verfahren in drei Schritten:

1. Einordnung der Testbilder in Kategorien aufgrund der (durch optische Betrachtung ermittelten) Trichteranzahl
2. Einordnung der Testbilder in Kategorien aufgrund der detektierten Trichteranzahl
3. Vergleich der Einordnung

Der Einfachheit halber wird im Folgenden oft von „bombardierte Bildern“ oder „Bilder, die bombardiert wurden“ gesprochen, wohl wissend, dass nicht die Bilder, sondern die auf den Bildern abgebildeten Gebiete bombardiert wurden.

Zunächst wurden die 30 Testbilder nach optischer Betrachtung (also ohne Abzählen der Trichter) in die drei Kategorien starke Bombardierung, mittlere Bombardierung und schwache Bombardierung eingeteilt<sup>3</sup>. Zur Kategorie schwach wurden auch die Bilder gezählt, die gar nicht bombardiert wurden.

Danach wurden für die Bilder jeder Kategorie die Anzahlen der detektierten Bombentrichter aufgelistet und der Median<sup>4</sup> gebildet, um einen Überblick über die mittlere Anzahl der Trichter in den Kategorien zu erhalten (Tabelle 4.10). Dann wurden Klassengrenzen gebildet. Sie wurden in

(a) schwach		(b) mittel		(c) stark	
Bildnr.	# T	Bildnr.	# T	Bildnr.	# T
261676	27	259796	210	255532	377
261680	31	266783	64	259853	529
261682	66	267016	54	266795	128
261686	39	267031	73	266798	205
266778	28	267077	38	266799	190
266781	22	277335	59	266879	123
268843	48	Median	62	267008	109
268845	42			267010	122
268851	54			270008	231
268854	66			277340	164
268855	42			Median	177
268858	72				
268867	59				
271617	37				
Median	42				

Tabelle 4.10: Anzahlen der detektierten Trichter auf den Testbildern in den drei Kategorien

die Mitte zwischen den jeweiligen Klassenmedianen gelegt<sup>5</sup>. Es ergeben sich folgende Intervalle für die Trichterzahlen: Kategorie schwach: 0-52, Kategorie mittel: 53-119, Kategorie stark: ab 120. Mit dieser Einteilung wurden die 30 Testbilder in die Kategorien schwach, mittel und stark eingeordnet.

Bei der Auswertung zeigte sich, dass 73% (absolut: 22) der Bilder der richtigen Kategorie zugeordnet wurden. Kein Bild mit schwacher Bombardierung wurde in die Kategorie stark bombardiert eingeordnet oder umgekehrt. Es gibt 7% (absolut: 2) Falschzuordnungen zwischen den Kategorien mittel und stark. Dabei wurde einmal ein Bild der Kategorie mittel in die Kategorie stark eingeordnet, und einmal anders herum. Bei 20% (absolut: 6) der Bilder wurde zwischen den Kategorien schwach und mittel falsch zugeordnet. Bei fünf davon wurden eigentlich schwach oder

<sup>3</sup>Die Bildung der Kategorien erfolgte ohne Rücksprache mit dem Kampfmittelbeseitigungsdienst und dient zunächst nur der Überprüfung der mit dem Algorithmus zu erzielenden Ergebnisse. Die Kategorien beziehen sich auf die Bombardierungsdichten der vorhandenen Testbilder und damit auf den ländlichen Raum. Ob der Kampfmittelbeseitigungsdienst diese Bilder in die gleichen Kategorien einordnen würde, oder ob dort mehr Gruppen unterschieden werden, ist zur Zeit nicht bekannt.

<sup>4</sup>Der Median ist hier dem Mittelwert vorzuziehen, da er robust gegen Ausreißer ist.

<sup>5</sup>Aufgrund von zeitlichen Beschränkungen und da es hier nur um einen ersten Überblick geht, wurde keine weitere Untersuchung zur besten Wahl der Klassengrenzen durchgeführt.

gar nicht bombardierte Bilder in die Kategorie mittel eingeordnet, bei einem war es umgekehrt.

Bei der Betrachtung der betreffenden Bilder stellte sich heraus, dass es bei Bildern mit starkem Kontrast oder starkem Schattenwurf, oder auch Waldgebieten viele kleine Strukturen gibt, die auch für den Menschen nur schlecht von Trichtern zu unterscheiden sind. Dadurch werden viele false positives detektiert, was die Einordnung in eine Kategorie mit stärkerer Bombardierung als eigentlich vorhanden ist, verursacht. Das Bild, das mit mittlerer Bombardierung in die Kategorie schwach eingeordnet wurde, ist im Vergleich dazu eher kontrastarm und von schlechterer Qualität. Außerdem enthält es viele „weiße“ Trichter, die mit dem Template nicht detektiert werden können (vgl. Abbildung 4.13).



(a) Bild der Kategorie schwach, das in Kategorie mittel eingeordnet wird

(b) Bild der Kategorie mittel, das in Kategorie schwach eingeordnet wird

Abbildung 4.13: Vergleich der Qualität zweier Bilder, die in falsche Kategorien eingeordnet wurden

Es ist eine wesentlich bessere Einordnung in die drei Kategorien zu erwarten, wenn die Anzahl der false positives durch Verwendung von mehreren Templates reduziert wird. Es ist zu bemerken, dass die Einordnung der Testbilder in Kategorien nach optischer Betrachtung durchaus auch mit Fehlern behaftet sein kann. Eine objektive Referenz erhielte man bei der vollständigen Annotation der Testbilder. Dann könnten eventuell auch mehr als die drei Kategorien (schwach, mittel und stark) gebildet werden. Bei der Durchführung der Trichterdetektion auf einer größeren Testmenge könnten außerdem die Klassengrenzen besser bestimmt werden.

Insgesamt kann man sagen, dass der implementierte Algorithmus vielversprechende Ergebnisse bei der Einordnung in Bombentrichter-dichte-Kategorien liefert, die sicher durch die oben erwähnten Optimierungen noch verbessert werden können.

# 5. Zusammenfassung und Ausblick

## Was erreicht wurde und was noch fehlt

### 5.1 Zusammenfassung

Der Kampfmittelbeseitigungsdienst der Bezirksregierung Arnsberg nutzt Luftbilder aus dem Zweiten Weltkrieg zur Detektion von Blindgängern. Die Arbeit der Auswerter könnte erleichtert werden, wenn sie eine Karte hätten, auf der die Dichte der Bombardierung dargestellt ist. Um diese Karte zu erstellen, ist ein Verfahren notwendig, das die Bombentrichter auf den Bildern automatisch detektiert. Dieses wurde in der Bachelorarbeit realisiert.

Da die Trichter sich in ihrer Gestalt und Größe stark unterscheiden, muss ein Ansatz zur Detektion gewählt werden, der mit diesen Variationen umgehen kann. Der Algorithmus führt eine Kandidatensuche mittels Kreuzkorrelation des Bildes mit einem repräsentativen Trichter-Template in verschiedenen Größen durch und klassifiziert die gefundenen Kandidaten anschließend. Die Klassifizierung erfolgt mit Hilfe der Wahrscheinlichkeitsdichte der Verteilungen der Klassen Trichter und Hintergrund. Um die Verteilungsparameter zu schätzen, ist die Reduktion des Merkmalsraums der Trainingsdaten mit einer Hauptkomponentenanalyse (PCA) und einer linearen Diskriminanzanalyse nach Fisher (LDA) und anschließender Projektion notwendig. In dieser Arbeit wurde das Verfahren mit einer Trichterklasse implementiert, es kann aber gut auf verschiedene Trichterklassen erweitert werden.

Zur Beurteilung der Ergebnisse wurde der Algorithmus auf den Trainingsbildern getestet und die Ergebnisse mit den Referenzdaten verglichen. Je nachdem ob vier oder fünf Templategrößen verwendet werden, können mit dem erstellten Template etwa 75% oder 80% der Trichter erfasst werden. Nach der Klassifikation werden mit dem implementierten Algorithmus je nach Konfiguration zwischen 70% und 64% der Trichter detektiert, dabei ist die Relevanz allerdings sehr gering. Maximal sind etwa 31% der als Trichter klassifizierten Bildausschnitte auch tatsächlich Bombentrichter.

Bei der Analyse der false positives auf Testbildern ergab sich, dass bestimmte Bildstrukturen, wie Hausdächer, Schattenwurf an Straßen, Texturen in Feldern oder Waldstrukturen immer wieder fälschlicherweise als Trichter klassifiziert werden. Bei der Untersuchung der nicht detektierten Bombentrichter konnten Trichterklassen abgeleitet werden, die mit dem erstellten Template nicht

detektiert werden. Mit den Testbildern wurde die Möglichkeit untersucht, die Bilder mit Hilfe der Bombentrichterdetektion in die Kategorien schwache, mittlere und starke Bombardierung einzuordnen. Hierbei wurden 73% der Bilder der richtigen Kategorie zugeordnet. Bei einer Steigerung der Relevanz und der Annotation weiterer Testbilder ist eine bessere Einordnung zu erwarten.

Insgesamt liegt mit dieser Arbeit ein vielversprechender Ansatz zur Bombentrichterdetektion mit großer Erweiterungsmöglichkeit vor.

## 5.2 Ausblick

Eine sinnvolle Erweiterung des Algorithmus ist die Verwendung von mehreren Templates zur Kandidatendetektion. Mögliche Trichterklassen, aus denen sie erstellt werden könnten, wurden in Abschnitt 4.4.2 abgeleitet. Es ist auch zu überlegen, ob der jeweils unterschiedlich ausgerichtete Schattenwurf der Trichter durch verschiedene Templates berücksichtigt wird. Dabei könnten durch Rotation zusätzliche Trainingsmuster generiert werden (Abschnitt 4.2.2). Wenn mehrere Templates verwendet werden, kann es auch sinnvoll sein, mit Ausschnitten aus den voll aufgelösten Bildern ( $11000 \times 12000$  Pixel) zu arbeiten. Die Bombentrichterausschnitte enthalten dann mehr Grauwerte aufgrund derer sie von anderen Bildstrukturen unterschieden werden könnten. Einige false positives könnten vermieden werden, wenn der Rand mit der Beschriftung nicht mit in die Detektion einbezogen würde.

Bei einer Verfeinerung des in dieser Arbeit dargestellten Algorithmus ist es möglich, die Schätzung der Verteilungen der Klassen zu optimieren. Dies könnte geschehen, indem erstens mehr Trainingsbeispiele verwendet werden<sup>1</sup>, um eine größere Stichprobe zu erhalten, und zweitens die Klassifikationsergebnisse untersucht werden, die sich bei der Annahme einer Gauß'schen Mischverteilung oder anderer Verteilungsklassen ergeben. Die Parameter der Gauß'schen Mischverteilung könnten zum Beispiel mit dem EM-Algorithmus, der in Bishop [2006] ausführlich dargestellt wird, bestimmt werden. Bei mehreren Trichterklassen könnten diese auch jeweils mit einer eigenen Verteilung modelliert werden.

Es ist ebenfalls denkbar, einen anderen Klassifikator zu verwenden, um die Klassifikationsergebnisse zu verbessern. Mit einer Support Vector Machine [Bishop, 2006] können beispielsweise auch nichtlineare Klassengrenzen gezogen werden. Schließlich könnte das Verfahren so angepasst werden, dass es das inkrementelle Lernen ermöglicht. Die Idee ist, dass ein Auswerter bei der Analyse der Klassifikation false positives und false negatives markiert, und diese dann zur Trainingsmenge der Hintergrundausschnitte bzw. Trichterausschnitte hinzugefügt werden. Das Modell zur Klassifizierung würde dann um die neuen Trainingsmustern erweitert und somit immer weiter verbessert.

Der Algorithmus könnte weiter optimiert werden, wenn die Luftbilder orientiert würden. Dadurch, dass die Aufnahmezeit der Bilder bekannt ist, könnte dann der Sonnenstand berechnet, der Schattenwurf modelliert und somit das Trichtermodell verbessert werden. Durch die Orientierung der Bilder wäre auch eine Zeitreihenanalyse möglich, indem die Bombentrichterdetektion auf mehreren Aufnahmen desselben Gebiets vor und nach der Bombardierung durchgeführt wird, und die Ergebnisse verglichen werden.

---

<sup>1</sup>Hierbei müsste das Speicherplatzproblem der PCA gelöst werden.



# Literaturverzeichnis

- Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer. [15, 17, 30, 52]
- Drauschke, M., Brunn, A., Kulschewski, K., and Förstner, W. (2007). Automatic dodging of aerial images. In Seyfert, E., editor, *Publikationen der DGPF: Von der Medizintechnik bis zur Planetenforschung - Photogrammetrie und Fernerkundung für das 21. Jahrhundert*, volume 16, pages 173–180. [1, 3, 20]
- Förstner, W. (1994). A framework for low level feature extraction. In *ECCV 1994*, number II in LNCS 801, pages 383–394. [6]
- Fukunaga, K. (1990). *Introduction to Statistical Pattern Recognition*. Academic Press. [18]
- Lindeberg, T. (1994). *Scale space theory in computer vision*. Kluwer Academic. [6]
- Lowe, D. G. (2004). Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110. [6]
- Lücke, C., Cruse, C., and Müterthies, A. (2007). Entwicklung und operationelle anwendung der halbautomatischen kriegsluftbildanalyse zur detektion von blindgängern. *Photogrammetrie, Fernerkundung, Geoinformation (PFG)*, 2007(7):495–502. [1]
- Vincent, L. and Soille, P. (1991). Watersheds in digital spaces: An efficient algorithm based on immersion simulations. *PAMI*, 13(6):583–598. [6]
- Wenzel, S. and Förstner, W. (2008a). Semi-supervised incremental learning of hierarchical appearance models. In *21st Congress of the International Society for Photogrammetry and Remote Sensing (ISPRS)*, Beijing, China. [7]
- Wenzel, S. and Förstner, W. (2008b). Semi-supervised prototype discovery for generating hierarchical appearance. *Interner Bericht*, 2008(7):4–13. [8, 54]

# Abbildungsverzeichnis

1.1	Ausschnitte der gleichen Auflösung aus einem Luftbild zeigen die Größenvariationen der Trichter . . . . .	3
1.2	Ausschnitte aus verschiedenen Bildern zeigen die Formvariationen der Trichter (nach dem Dodging) . . . . .	3
1.3	Trichterähnliche Objekte . . . . .	4
1.4	Ergebnis der Suche nach Trichtern mittels Kreuzkorrelation. Aus Wenzel and Förstner [2008b]. . . . .	8
2.1	Erstellung der Templates für die Kandidatensuche mittels Kreuzkorrelation . . .	10
2.2	Ablauf der Dimensionsreduktion des Merkmalsraums für die Trainingsmuster . .	12
2.3	Ablauf der Bestimmung der Verteilungsparameter der Klassen Trichter und Hintergrund . . . . .	13
2.4	Ablauf der Arbeitsphase . . . . .	14
2.5	Projektion mit der Hauptkomponentenanalyse . . . . .	15
2.6	blau: Punkte der Klasse $\mathcal{C}_1$ - rot: Punkte der Klasse $\mathcal{C}_2$ . . . . .	17
2.7	blau: Verteilung der Punkte in Klasse $\mathcal{C}_1$ - rot: Verteilung der Punkte in Klasse $\mathcal{C}_2$	17
3.1	Ein Luftbild vor der Anwendung des Dodgings (links) und danach (rechts). . . .	20
3.2	Die Durchmesser (in Pixel) aller annotierten Bombentrichter in 24 Bildern . . . .	23
3.3	Mittlerer Trichter der Größe $17 \times 17$ Pixel . . . . .	25
3.4	Summe der verworfenen Eigenwerte in Abhängigkeit von der Dimension $K$ . . . .	29
3.5	Histogramm der 3600 Bildausschnitte nach der LDA, rot: Hintergrund - blau: Trichter . . . . .	31
3.6	Histogramm der 900 Trichterausschnitte und 2700 Hintergrundausschnitte nach der LDA . . . . .	31
4.1	Mittlerer Trichter der Größe $17 \times 17$ Pixel . . . . .	39
4.2	False positives an einer Strasse (Bild 268843) . . . . .	42
4.3	False positives an Hausdächern (Bild 266783) . . . . .	43
4.4	False positives auf einem Feld (Bild 268851) . . . . .	43
4.5	False positives im Wald (Bild 268855) . . . . .	44
4.6	False positives, die nicht eindeutig zu interpretieren sind (Bild 266781) . . . .	44
4.7	False positives am Rand (Bild 266879) . . . . .	45
4.8	„Weiße Bombentrichter“ als false negatives (Bild 259853) . . . . .	45
4.9	„Schwarze Bombentrichter“ als false negatives (Bild 255532) . . . . .	46
4.10	Kontrastarme Trichter als false negatives (Bild 267008) . . . . .	47
4.11	Kleine Trichter als false negatives (Bild 267008) . . . . .	47

4.12	Uninterpretierbare als false negatives (Bild 259853) . . . . .	48
4.13	Vergleich der Qualität zweier Bilder, die in falsche Kategorien eingeordnet wurden	50
B.1	False positives im Wald . . . . .	59
B.2	False positives an Strassen . . . . .	60
B.3	False positives an Hausdächern . . . . .	61
B.4	False positives auf Feldern . . . . .	62
B.5	False positives, die nicht eindeutig zu interpretieren sind . . . . .	63
B.6	„Weiße Trichter“ als false negatives . . . . .	64
B.7	„Schwarze Trichter“ als false negatives . . . . .	65
B.8	Kontrastarme als false negatives . . . . .	66
B.9	Kleine Trichter als false negatives . . . . .	67
B.10	Nicht interpretierbare false negatives . . . . .	68

# Tabellenverzeichnis

3.1	Statistik über die annotierten Bombentrichter . . . . .	22
3.2	Minimale und maximale Trichtergrößen, die mit den Templates erreichbar sind .	25
3.3	Fehlklassifikationsraten bei den 3600 Trainingsmustern . . . . .	32
3.4	Fehlklassifikationsraten bei unterschiedlich großer Umgebung auf den Ausschnitten	33
3.5	Mittlere Fehlklassifikationsraten bei unterschiedlich großer Umgebung auf den Ausschnitten . . . . .	33
4.1	Beispiel für eine Konfusionstabelle bei der Kandidatensuche . . . . .	34
4.2	Beispiel für eine Konfusionstabelle bei der Klassifikation mit absoluten Werten .	35
4.3	Beispiel für eine Konfusionstabelle bei der Klassifikation mit relativen Werten . .	35
4.4	Ergebnis der Kandidatensuche (mit verschiedener Templatezahl) nach Vergleich mit dem Trainingsbild . . . . .	37
4.5	Konfusionstabellen der Kandidatensuche für beide Gruppen bei verschiedener Templatezahl . . . . .	37
4.6	Konfusionstabellen der Kandidatensuche bei rotationsinvariantem und nicht rota- tionsinvariantem Template . . . . .	38
4.7	Absolute Klassifikationsergebnisse bei allen Trainingsbildern mit verschiedener Templatezahl . . . . .	40
4.8	Konfusionstabellen für die Klassifikation bei Suche mit allen Templates . . . . .	41
4.9	Konfusionstabellen für die Klassifikation bei Suche ohne Templategröße 9 . . . . .	41
4.10	Anzahlen der detektierten Trichter auf den Testbildern in den drei Kategorien . .	49

# Anhang A

## Notation

### *Erstellung des Templates*

$\mathbf{T}'_t$	$t = 1, \dots, T$	Trichterausschnitte für die Templateerstellung in verschiedenen Größen
$\mathbf{T}_t \in \mathbb{R}^{I \times J}$		skalierte Trichterausschnitte
$\mathbf{M} = \frac{1}{T} \sum_{t=1}^T \mathbf{T}_t$		Template für die Kreuzkorrelation
$\mathbf{M}_1 \dots \mathbf{M}_S$		Skalieren von $\mathbf{M}$ in $S$ Größen.

### *Auswahl der Trainingsmuster*

$\mathbf{T}'_m$	$m = 1, \dots, M$	Trichterausschnitte in verschiedenen Größen
$\mathbf{H}'_l$	$l = 1, \dots, L$	Hintergrundausschnitte in verschiedenen Größen
$\mathbf{T}_m \in \mathbb{R}^{I \times J}$		skalierte Trichterausschnitte
$\mathbf{H}_l \in \mathbb{R}^{I \times J}$		skalierte Hintergrundausschnitte
$\mathbf{t}_m \in \mathbb{R}^{D \times 1}$		vektorierte Trichterausschnitte
$\mathbf{h}_l \in \mathbb{R}^{D \times 1}$		vektorierte Hintergrundausschnitte
$N = M + L$		Anzahl aller Bildausschnitte

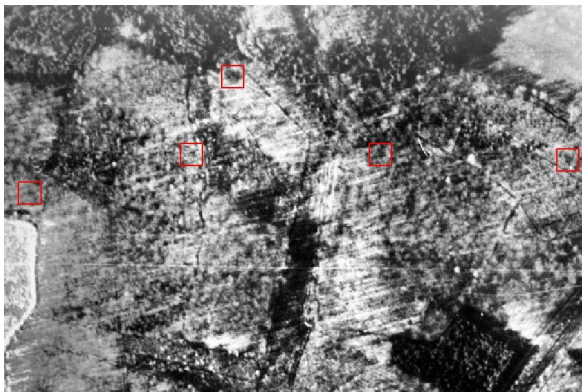
### *Dimensionsreduktion des Merkmalsraums mit PCA und LDA und Schätzung der Parameter der Wahrscheinlichkeitsdichtefunktionen*

$\mathbf{X} \in \mathbb{R}^{D \times N}$	Matrix mit den Trainingsdaten
$\boldsymbol{\mu}_{\mathbf{x}} = \frac{1}{N} \sum_{n=1}^N \mathbf{x}_n \in \mathbb{R}^{D \times 1}$	Mittelwert aller Trainingsdaten
$\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_D] \in \mathbb{R}^{D \times D}$	Matrix mit den $D$ Eigenvektoren aus der PCA
$\lambda_d$	zu den $\mathbf{u}_d$ gehörende Eigenwerte
$\mathbf{U}_K = [\mathbf{u}_1, \dots, \mathbf{u}_K] \in \mathbb{R}^{D \times K}$	Matrix mit den ersten $K$ Eigenvektoren
$\mathbf{y}_n = \mathbf{U}_K^T \cdot (\mathbf{x}_n - \boldsymbol{\mu}_{\mathbf{x}})$	Projektion der Trainingsmuster $\mathbf{x}_n$ in den $K$ -dimensionalen Raum
$\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_N] \in \mathbb{R}^{K \times N}$	Matrix mit den in den PCA-Raum projizierten Trainingsdaten
$\mathbf{V} = [\mathbf{v}_1, \dots, \mathbf{v}_R] \in \mathbb{R}^{K \times R}$	Projektionsmatrix aus der LDA

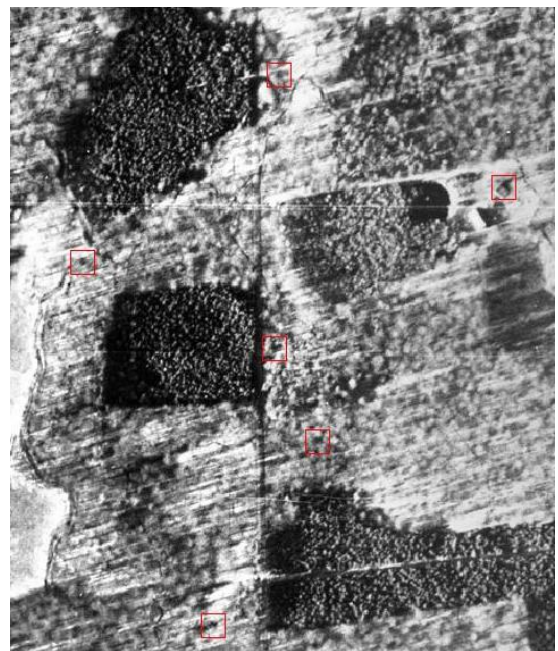
$z_n = \mathbf{V}^T \cdot \mathbf{y}_n$	Projektion der Trainingsmuster $\mathbf{y}_n$ in den $R$ -dimensionalen Raum
$\mathbf{Z} = [z_1, \dots, z_N] \in \mathbb{R}^{R \times N}$	Matrix mit den in den LDA-Raum projizierten Trainingsdaten
$\mathbf{W}^T = \mathbf{V}^T \mathbf{U}_K^T \in \mathbb{R}^{R \times D}$	zusammengesetzte Projektionsmatrix
$z_n = \mathbf{W}^T \cdot (\mathbf{x}_n - \boldsymbol{\mu}_x)$	Projektion der Trainingsmuster $\mathbf{x}_n$ in den $R$ -dimensionalen Raum
$z_m, z_l \in \mathbb{R}^{R \times 1}$	in den $R$ -dimensionalen Raum projizierte
	Trichter-/Hintergrundausschnitte
$\mathcal{P}_T(\cdot)$	Wahrscheinlichkeitsdichtefunktion der Verteilung der Trichter
$\mathcal{P}_H(\cdot)$	Wahrscheinlichkeitsdichtefunktion der Verteilung des Hintergrunds
<i>Arbeitsphase</i>	
$\mathbf{B} \in \mathbb{R}^{U \times V}$	Grauwertbild, in dem Trichter gefunden werden sollen.
$\mathbf{G}'_p$	durch Maximumssuche gefundene Kandidaten
$\mathbf{G}_p \in \mathbb{R}^{I \times J}$	skalierte Kandidaten
$\mathbf{g}_p \in \mathbb{R}^{D \times 1}$	vektorierte Kandidaten
$\bar{\mathbf{g}}_p = \mathbf{W}^T \cdot (\mathbf{g}_p - \boldsymbol{\mu}_x)$	in den $R$ -dimensionalen Raum projizierte Kandidaten
$\mathcal{P}_T(\bar{\mathbf{g}}_p)$	Wahrscheinlichkeitsdichte der Verteilung der Trichter an der Stelle $\bar{\mathbf{g}}_p$
$\mathcal{P}_H(\bar{\mathbf{g}}_p)$	Wahrscheinlichkeitsdichte der Verteilung des Hintergrunds an der Stelle $\bar{\mathbf{g}}_p$

## Anhang B

## Bilder



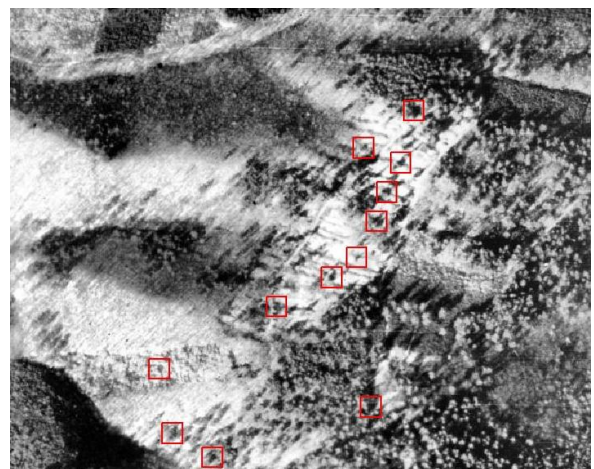
(a) Bild 268855



(b) Bild 268851



(c) Bild 268858



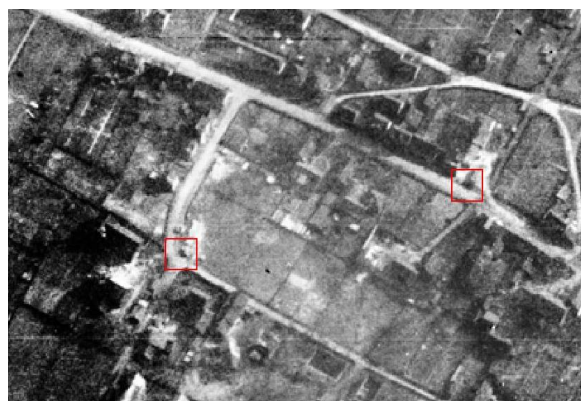
(d) Bild 274840

Abbildung B.1: False positives im Wald





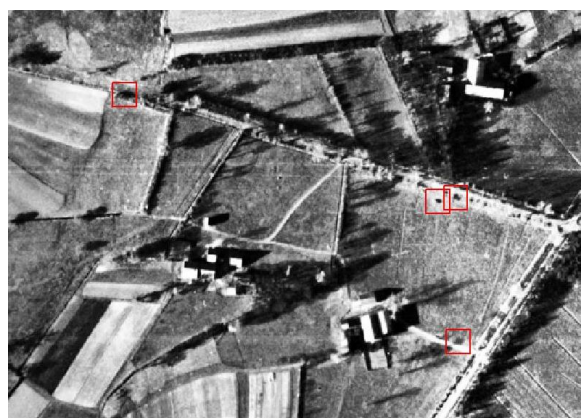
(a) Bild 266781



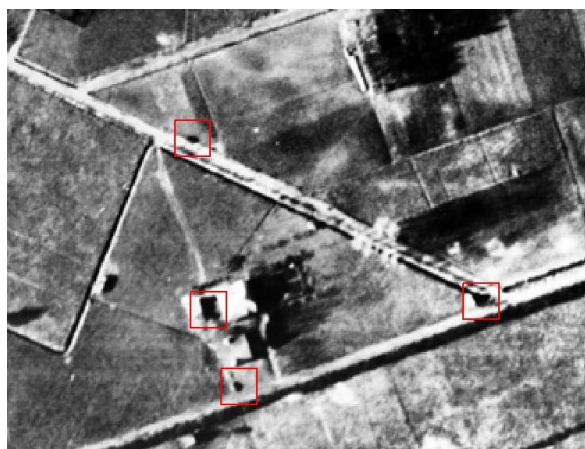
(b) Bild 266798



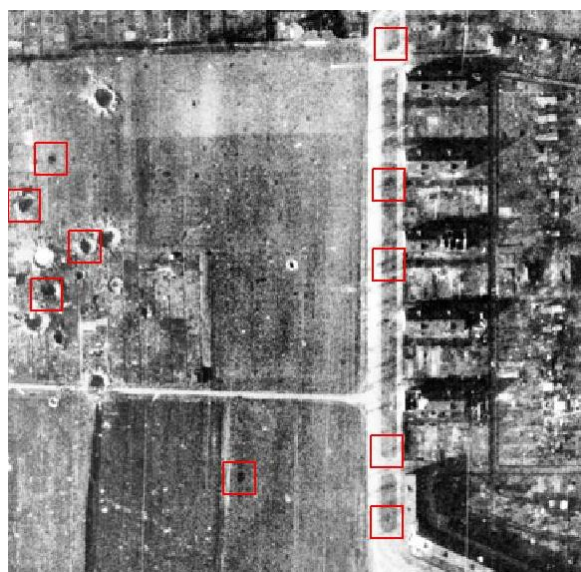
(c) Bild 268843



(d) Bild 268854



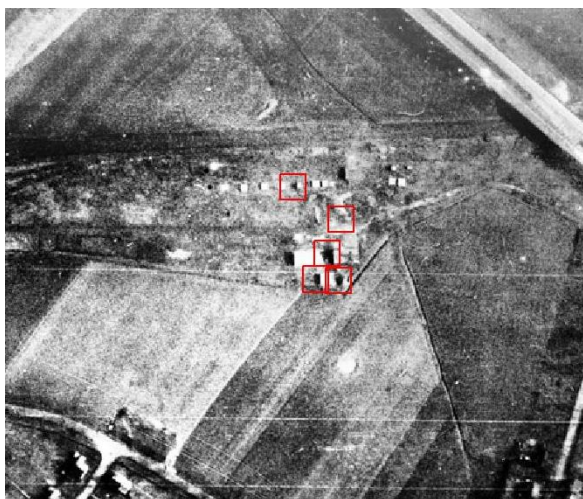
(e) Bild 268867



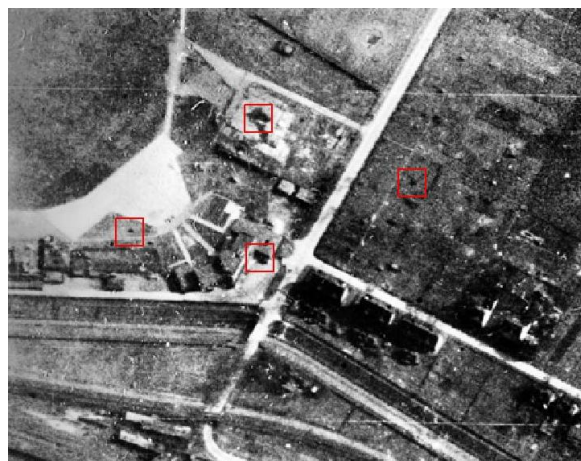
(f) Bild 277340

Abbildung B.2: False positives an Strassen

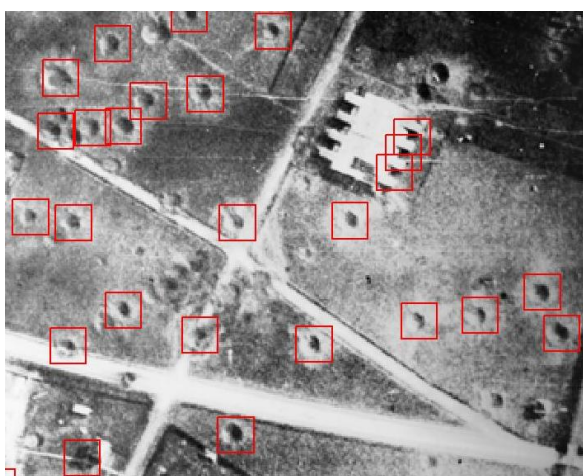




(a) Bild 266783



(b) Bild 266795



(c) Bild 266799



(d) Bild 277340



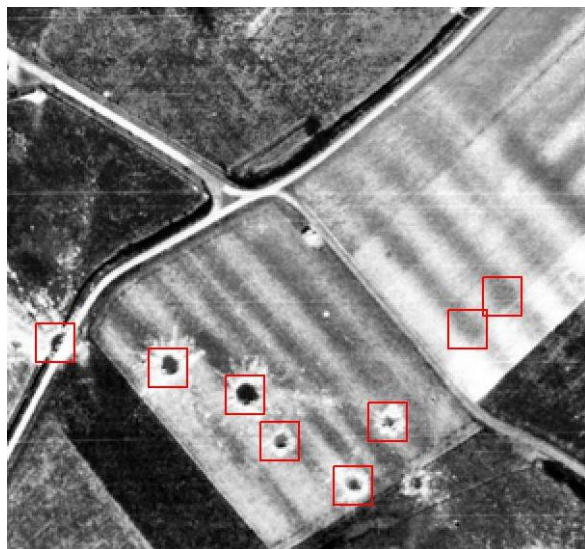
(e) Bild 268867



(f) Bild 271617

Abbildung B.3: False positives an Hausdächern





(a) Bild 259796



(b) Bild 261676



(c) Bild 261680



(d) Bild 268845



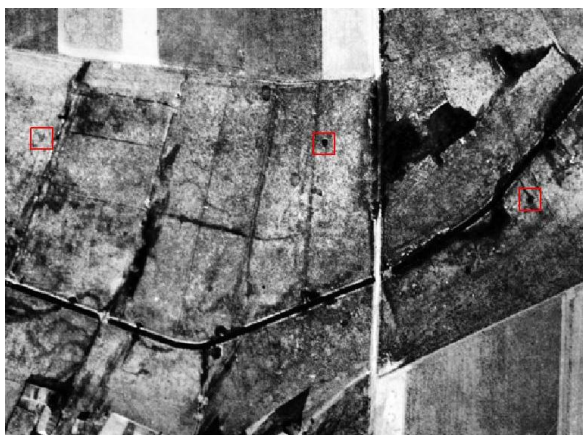
(e) Bild 268855



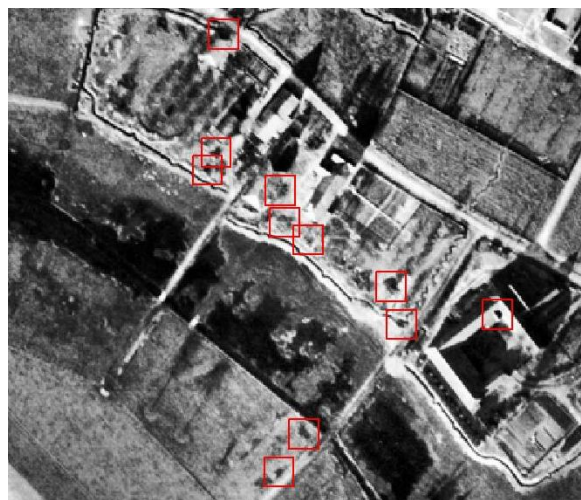
(f) Bild 268858

Abbildung B.4: False positives auf Feldern

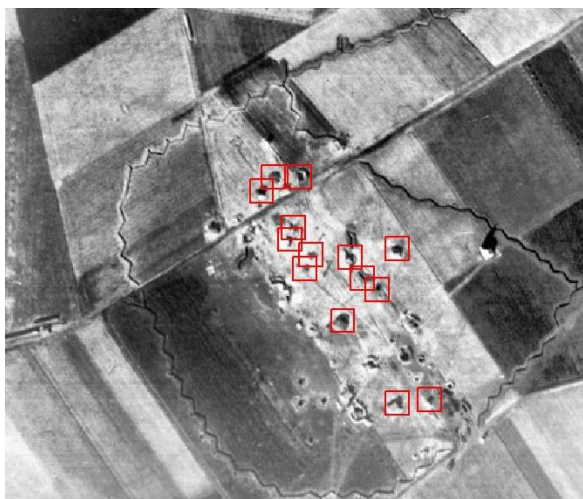




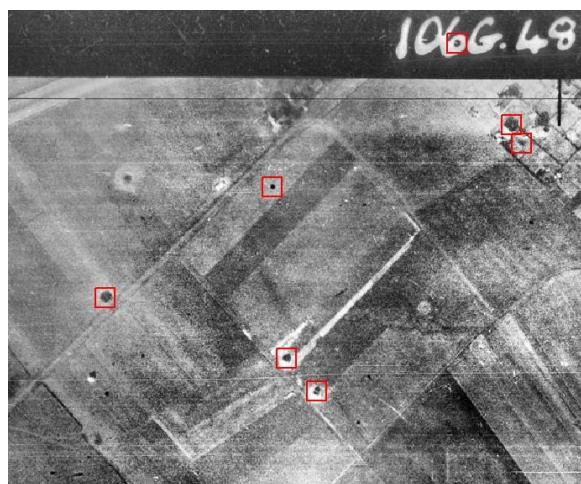
(a) Bild 261680



(b) Bild 261682



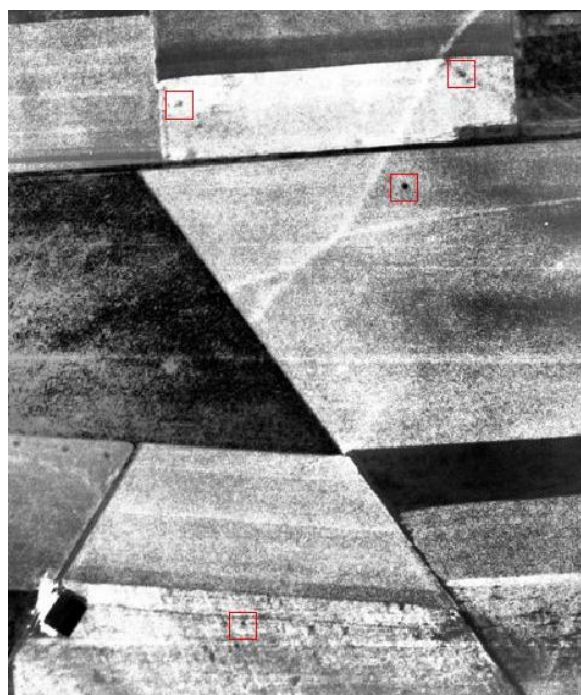
(c) Bild 261682



(d) Bild 267077



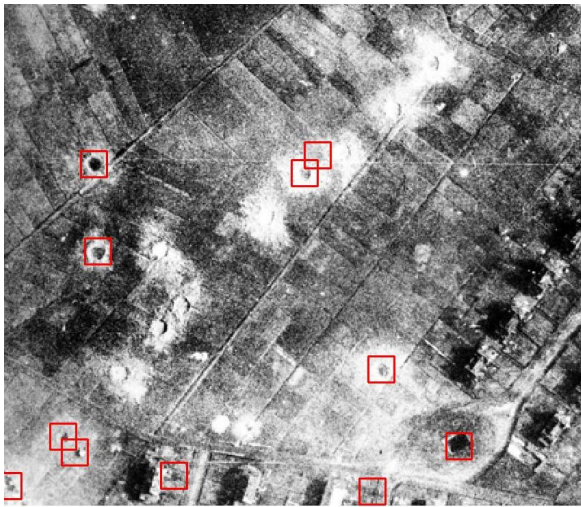
(e) Bild 270008



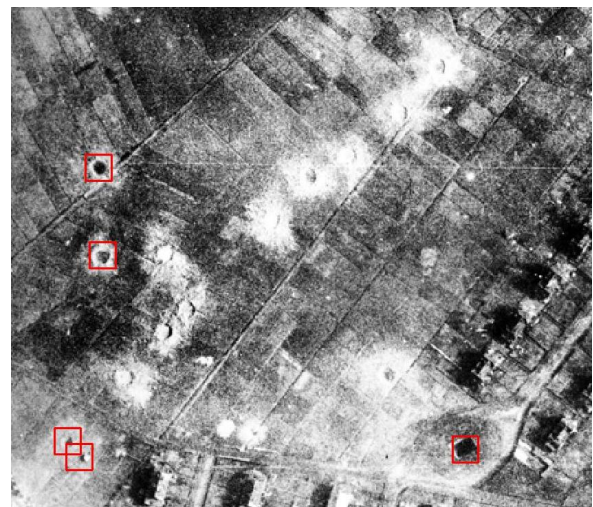
(f) Bild 271617

Abbildung B.5: False positives, die nicht eindeutig zu interpretieren sind

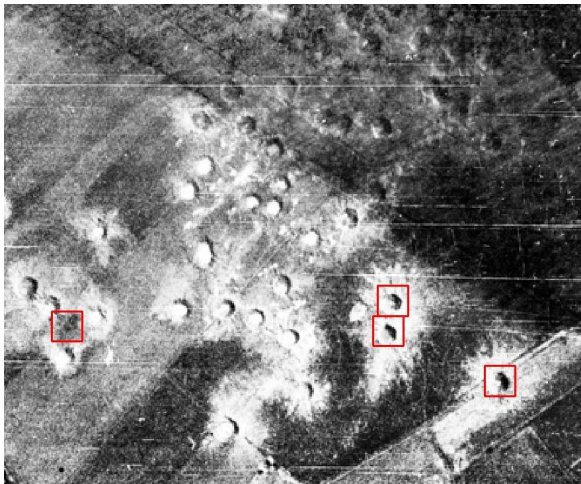




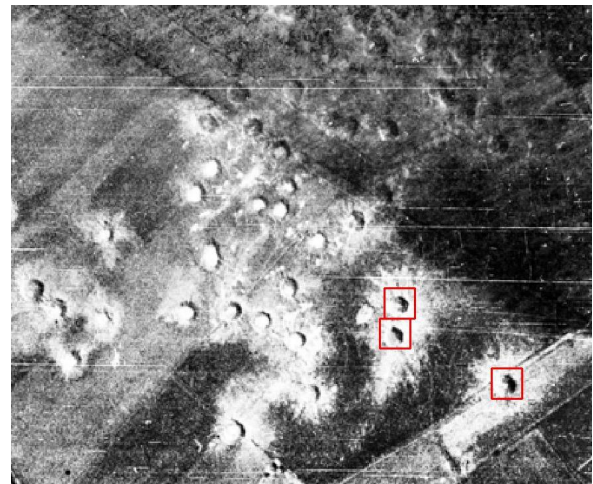
(a) Bild 266783 - nach Kandidatensuche



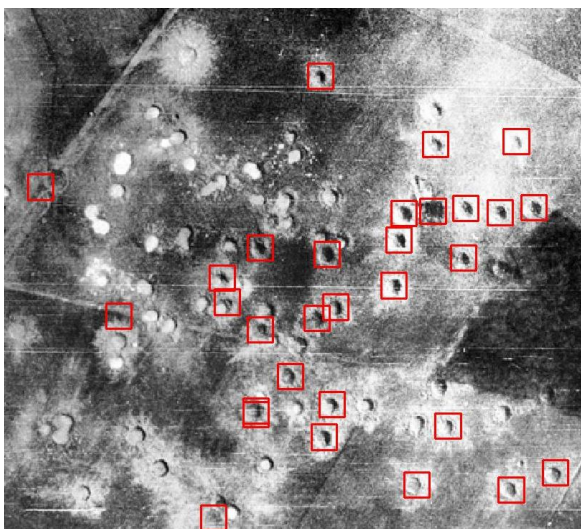
(b) Bild 266783 - nach Klassifikation



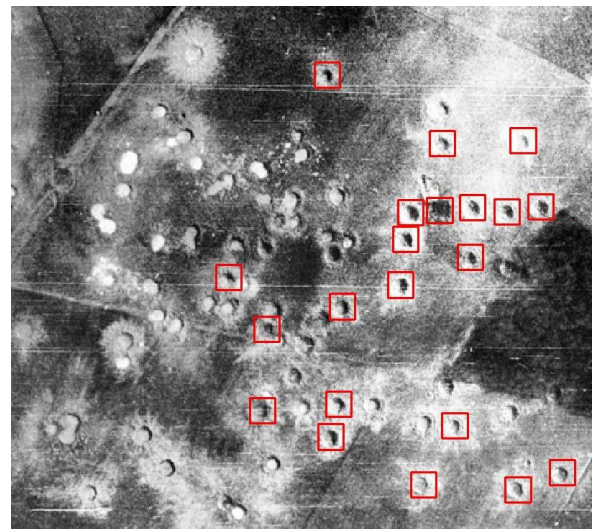
(c) Bild 266879 - nach Kandidatensuche



(d) Bild 266879 - nach Klassifikation



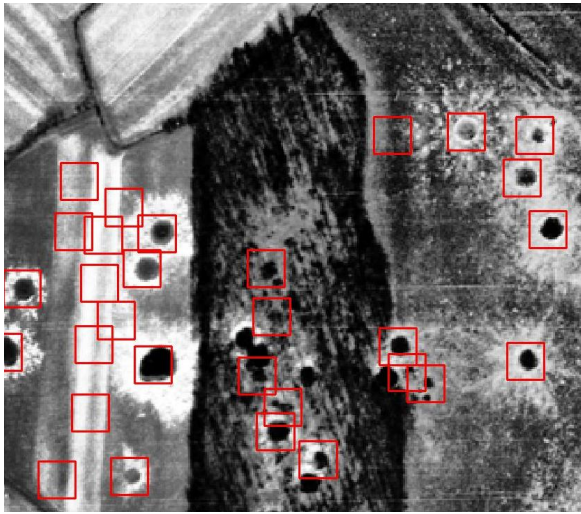
(e) Bild 266879 - nach Kandidatensuche



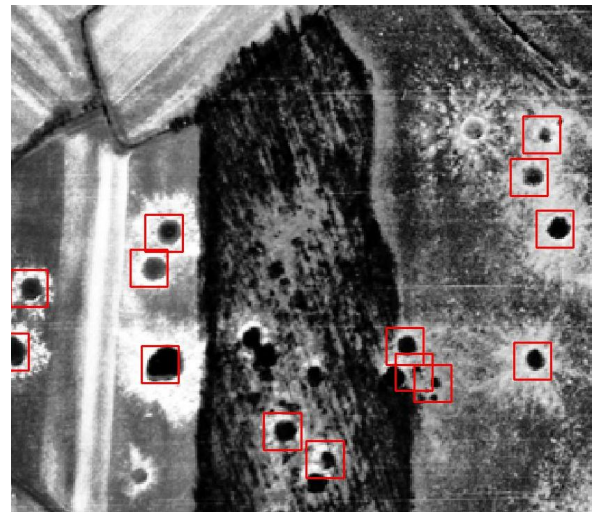
(f) Bild 266879 - nach Klassifikation

Abbildung B.6: „Weiße Trichter“ als false negatives





(a) Bild 266783 - nach Kandidatensuche



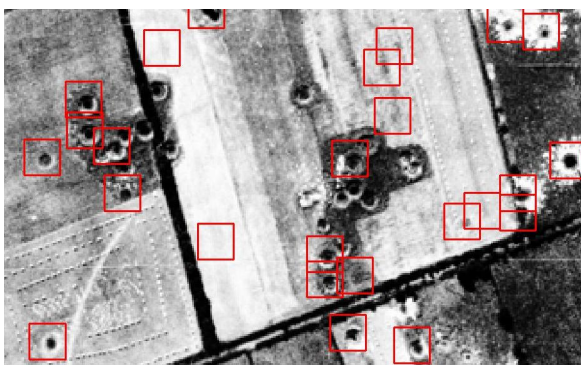
(b) Bild 266783 - nach Klassifikation



(c) Bild 266795 - nach Kandidatensuche



(d) Bild 266795 - nach Klassifikation



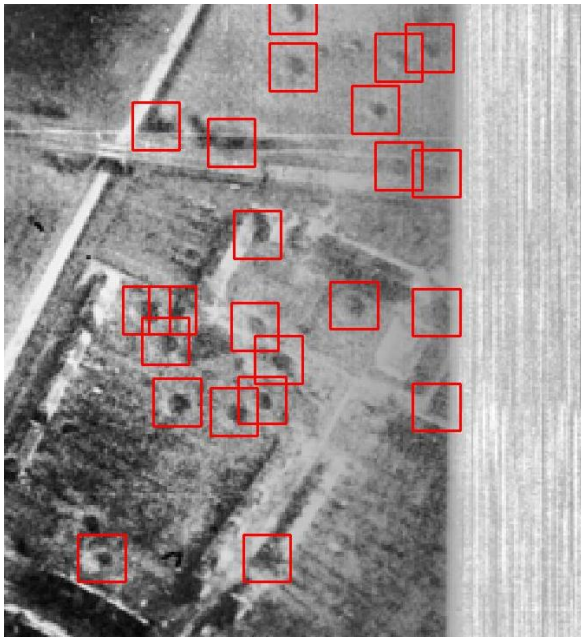
(e) Bild 270008 - nach Kandidatensuche



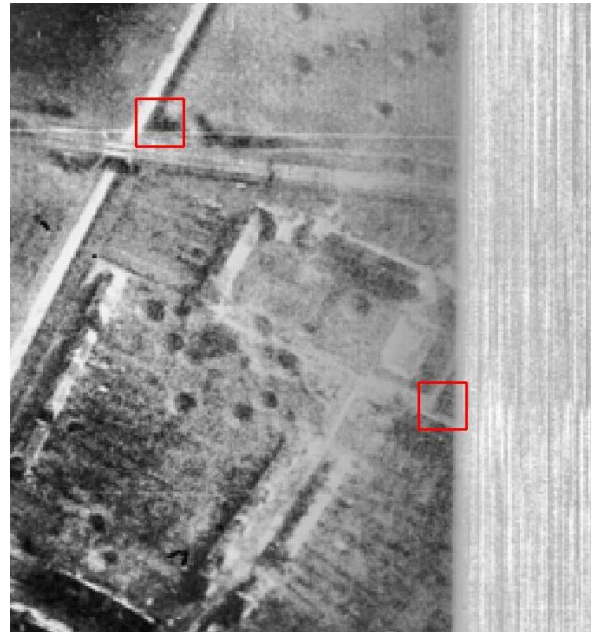
(f) Bild 270008 - nach Klassifikation

Abbildung B.7: „Schwarze Trichter“ als false negatives

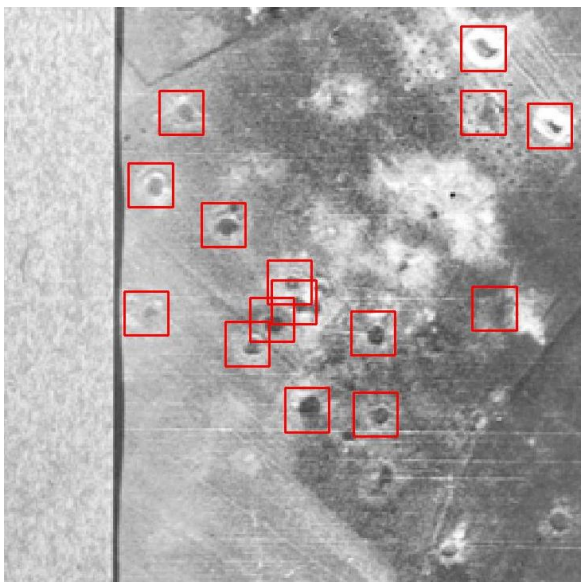




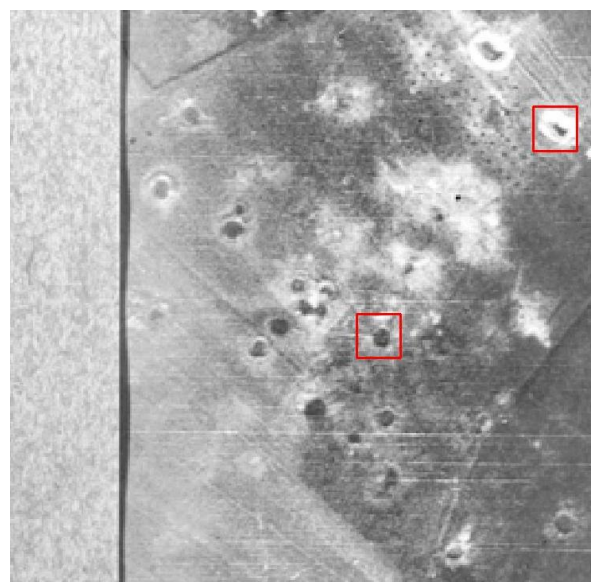
(a) Bild 266799 - nach Kandidatensuche



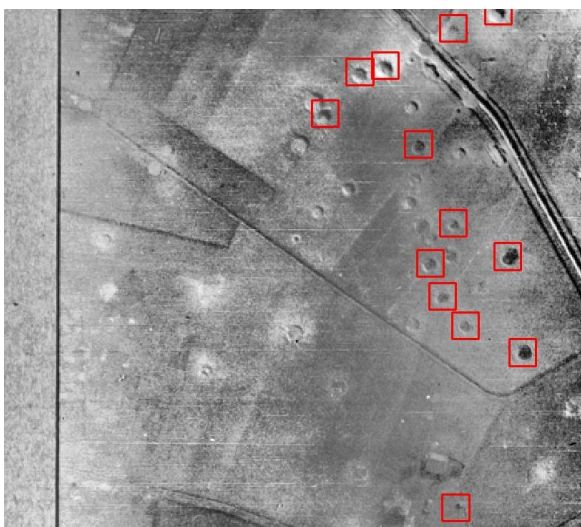
(b) Bild 266799 - nach Klassifikation



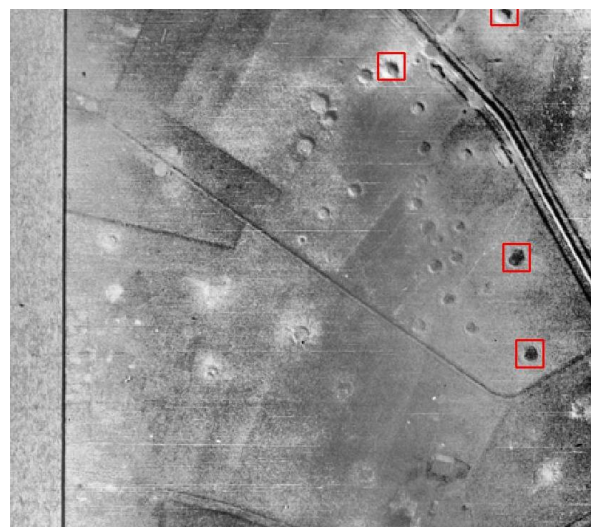
(c) Bild 267008 - nach Kandidatensuche



(d) Bild 267008 - nach Klassifikation



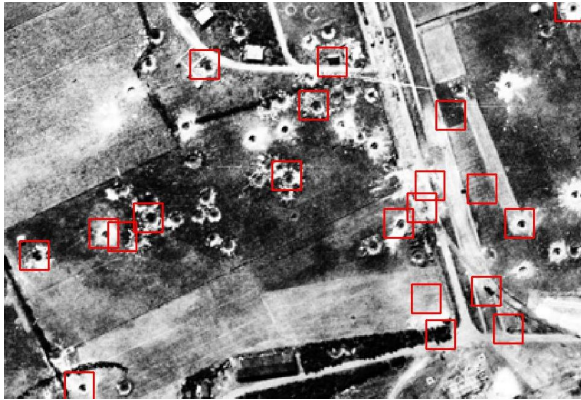
(e) Bild 267031 - nach Kandidatensuche



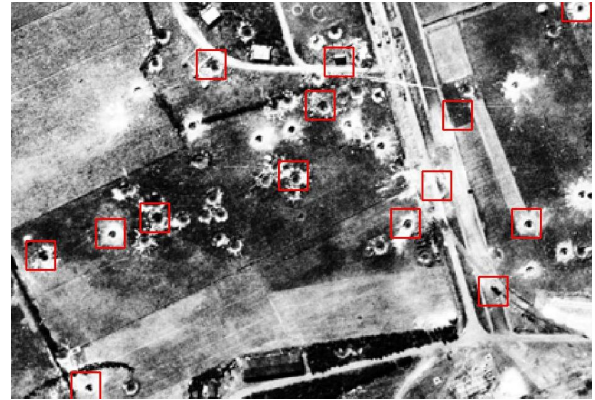
(f) Bild 267031 - nach Klassifikation

Abbildung B.8: Kontrastarme als false negatives

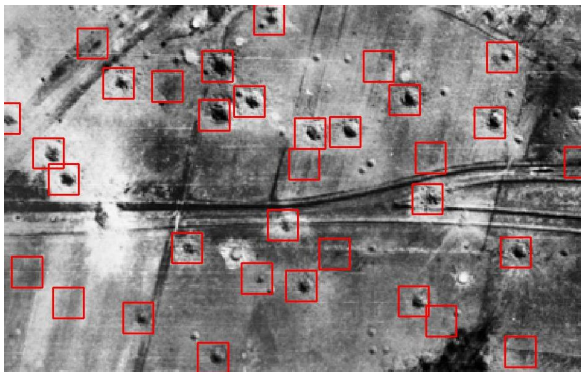




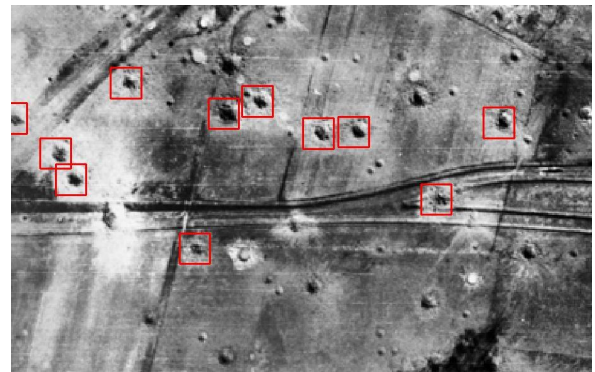
(a) Bild 270008 - nach Kandidatensuche



(b) Bild 270008 - nach Klassifikation



(c) Bild 259853 - nach Kandidatensuche



(d) Bild 259853 - nach Klassifikation



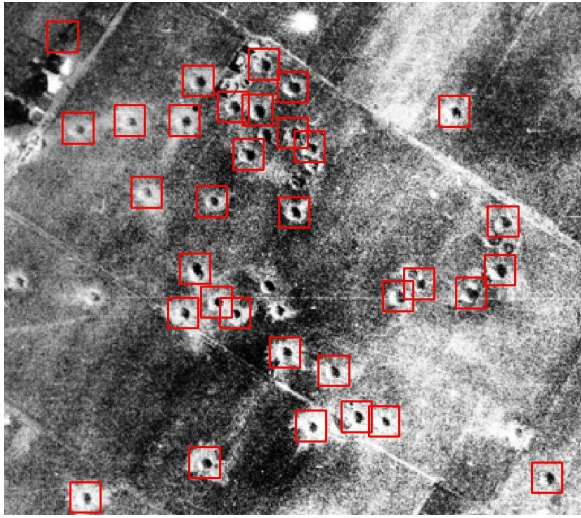
(e) Bild 267031 - nach Kandidatensuche



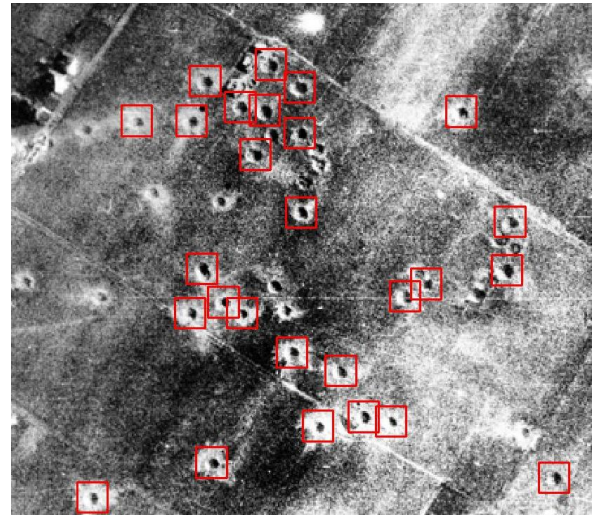
(f) Bild 267031 - nach Klassifikation

Abbildung B.9: Kleine Trichter als false negatives

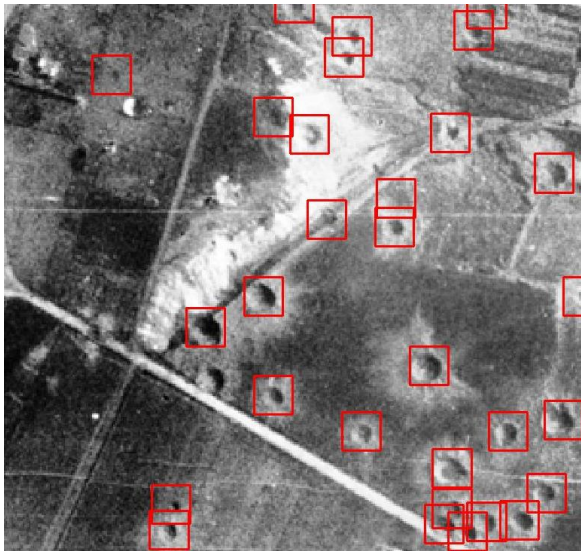




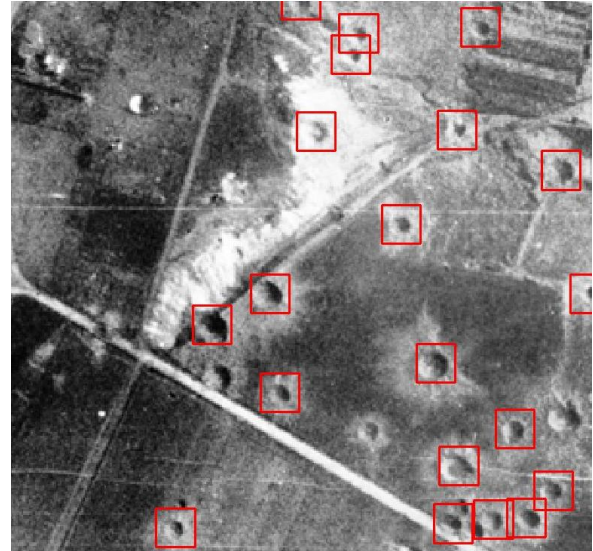
(a) Bild 266798 - nach Kandidatensuche



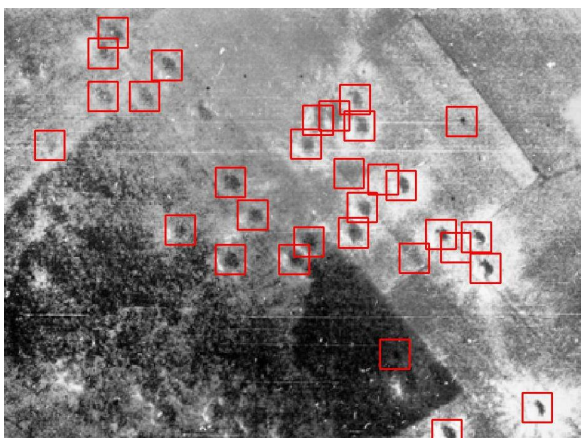
(b) Bild 266798 - nach Klassifikation



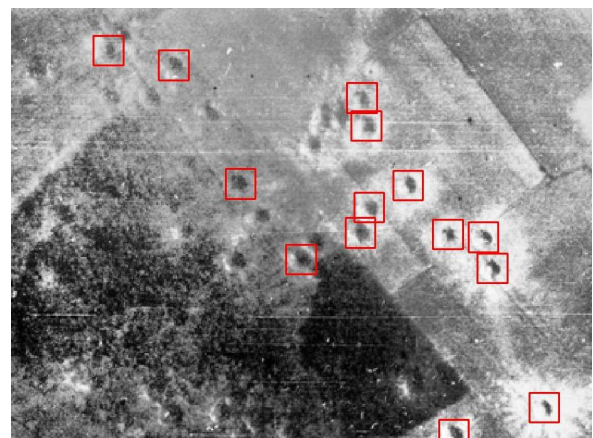
(c) Bild 266799 - nach Kandidatensuche



(d) Bild 266799 - nach Klassifikation



(e) Bild 266879 - nach Kandidatensuche



(f) Bild 266879 - nach Klassifikation

Abbildung B.10: Nicht interpretierbare false negatives



Anhang C

Hausarbeit zum Oberseminar

# Automatische Detektion von Bombentrichtern

von Laura Jensen

## 1 Aufgabenstellung

Während des zweiten Weltkriegs machten die alliierten Streitkräfte über 300000 Luftbilder<sup>1</sup> von Nordrhein-Westfalen. Heute werden diese Bilder unter anderem vom Kampfmittelbeseitigungsdienst genutzt, um Blindgänger zu lokalisieren. Im Jahr 2006 wurden allein in Nordrhein-Westfalen knapp 1100 nicht detonierte Bomben entschärft<sup>2</sup>. Da die manuelle Untersuchung der Bilder aufgrund der großen Menge sehr aufwendig ist, ist es wünschenswert, die Suche nach Bombentrichtern in den Luftbildern zu automatisieren. Aufgabe der Bachelorarbeit ist es, ein Verfahren zu entwickeln und umzusetzen, dass die automatische Detektion von Bombentrichtern im ländlichen Raum realisiert.

Hierfür stehen Luftbilder zur Verfügung, die sich in ihrer Qualität und ihrem Maßstab unterscheiden. Globale Helligkeitsunterschiede, die aus der Aufnahmesituation resultieren, wurden vorher mit Hilfe eines bereits bestehenden Verfahrens, das auf abschnittsweiser Histogrammverebnung beruht, weitgehend angepasst. Damit besteht eine globale Vergleichbarkeit der Bilder bezüglich ihrer Grauwertverteilung.

Die auf den Bildern erkennbaren Bombentrichter unterscheiden sich, wie in Abbildung 1 und 2 dargestellt, in Größe und Form. Das Aussehen der nahen Umgebung der Einschlaglöcher variiert

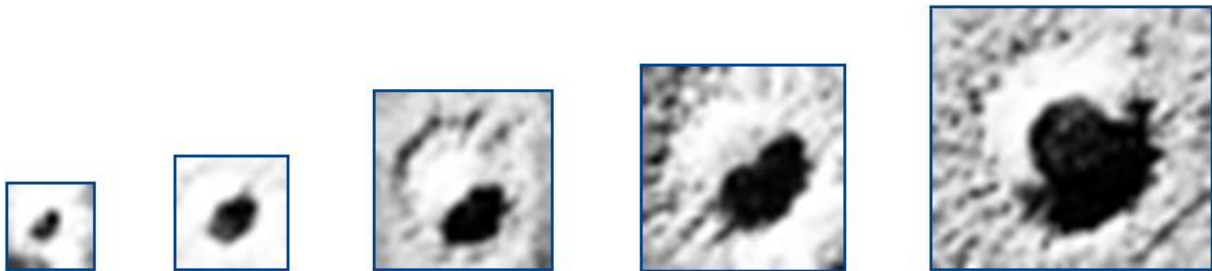


Abbildung 1: Ausschnitte der gleichen Auflösung aus einem Luftbild zeigen die Größenvariationen der Trichter

aufgrund der unterschiedlichen Ausprägung des Erdauswurfs, der bei ihrer Detonation entstand, ebenfalls. Des Weiteren ähneln die Bombentrichter anderen Objekten im Bild (Abbildung 3). Bei der Detektion sollen die Trichter lediglich lokalisiert, nicht aber nach ihrer Art oder Größe unterschieden werden.

---

<sup>1</sup>siehe Homepage des Innenministeriums NRW, <http://www.im.nrw.de/sch/725.htm>, zuletzt aufgerufen am 14.04.2007

<sup>2</sup>siehe ebda.

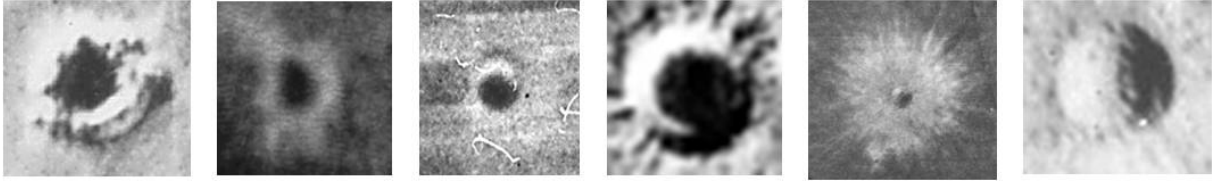


Abbildung 2: Ausschnitte aus verschiedenen Bildern zeigen die Formvariationen der Trichter

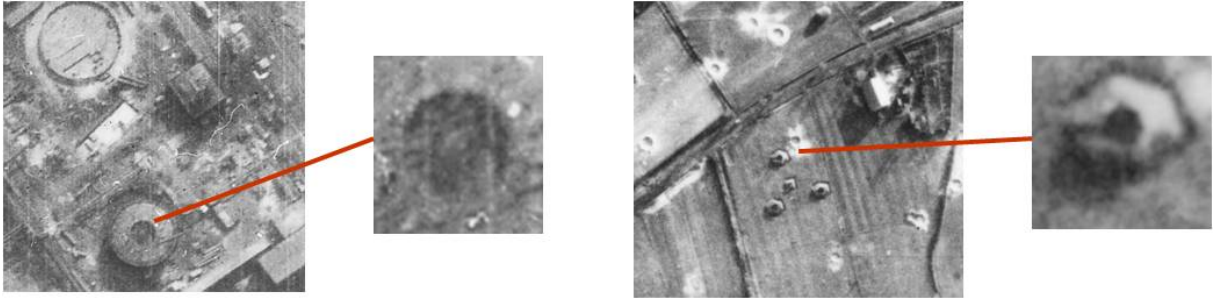


Abbildung 3: Objekte in Bildern, die Trichtern ähneln

## 2 Konzept und Methoden

### 2.1 Beschreibung des Gesamtkonzepts

Die Idee ist, mögliche Trichterpositionen im Bild mittels Kreuzkorrelation zu finden. Als Templates sollen mehrere repräsentative Bilder von Trichtern in verschiedenen Größen genutzt werden, um möglichst viele Trichter zu erfassen. An den gefundenen Positionen werden dann Bildfenster ausgeschnitten und diese als Trichter oder Hintergrund klassifiziert.

Abbildung 4 zeigt das Gesamtkonzept der Bombentrichterdetektion, das in der Bachelorarbeit umgesetzt werden soll. Es besteht aus dem Trainingsteil, in dem die Templates für die Kreuzkorrelation erstellt werden und die Trennfunktion für die Klassifikation bestimmt wird, und dem Teil der Trichterdetektion, in dem die Kreuzkorrelation und die Klassifikation mit Testbildern durchgeführt wird. Im Folgenden wird das Gesamtkonzept genauer erläutert:

Aus einer Trainingsmenge von Luftbildern wird zunächst eine Menge von quadratischen Bildfenstern mit Trichtern und Hintergrund ausgeschnitten. Die Fenster des Hintergrunds werden dabei zufällig gewählt. Die Ausschnitte werden auf die gleiche Größe (z.B. 31 x 31 Pixel) skaliert und als Vektoren in einem 961-dimensionalen Raum aufgefasst. Es wird nun eine Hauptkomponentenanalyse<sup>3</sup> durchgeführt, die im Abschnitt 2.2.1 näher erläutert wird.

Die ersten  $M$  961-dimensionalen Eigenvektoren, die man bei der Hauptkomponentenanalyse erhält, werden als „Eigenbilder“ interpretiert. Sie zeigen die Hauptmerkmale der Bildausschnitte. Da die Ausschnitte des Hintergrunds aufgrund ihrer Inhomogenität nichts zu diesen Hauptmerkmalen beitragen, stellen die „Eigenbilder“ die Hauptausprägungen der Trichter dar, und können als Templates für die Kreuzkorrelation verwendet werden. Hierfür werden sie auf  $K$  Größen skaliert, um den Größenvariationen der Bombentrichter gerecht zu werden.

Die ersten  $M$  Eigenvektoren der Hauptkomponentenanalyse werden außerdem dazu genutzt, die skalierten Bildausschnitte in einen  $M$ -dimensionalen Raum zu projizieren. Mit den projizierten

<sup>3</sup>engl.: Principle Components Analysis (PCA)

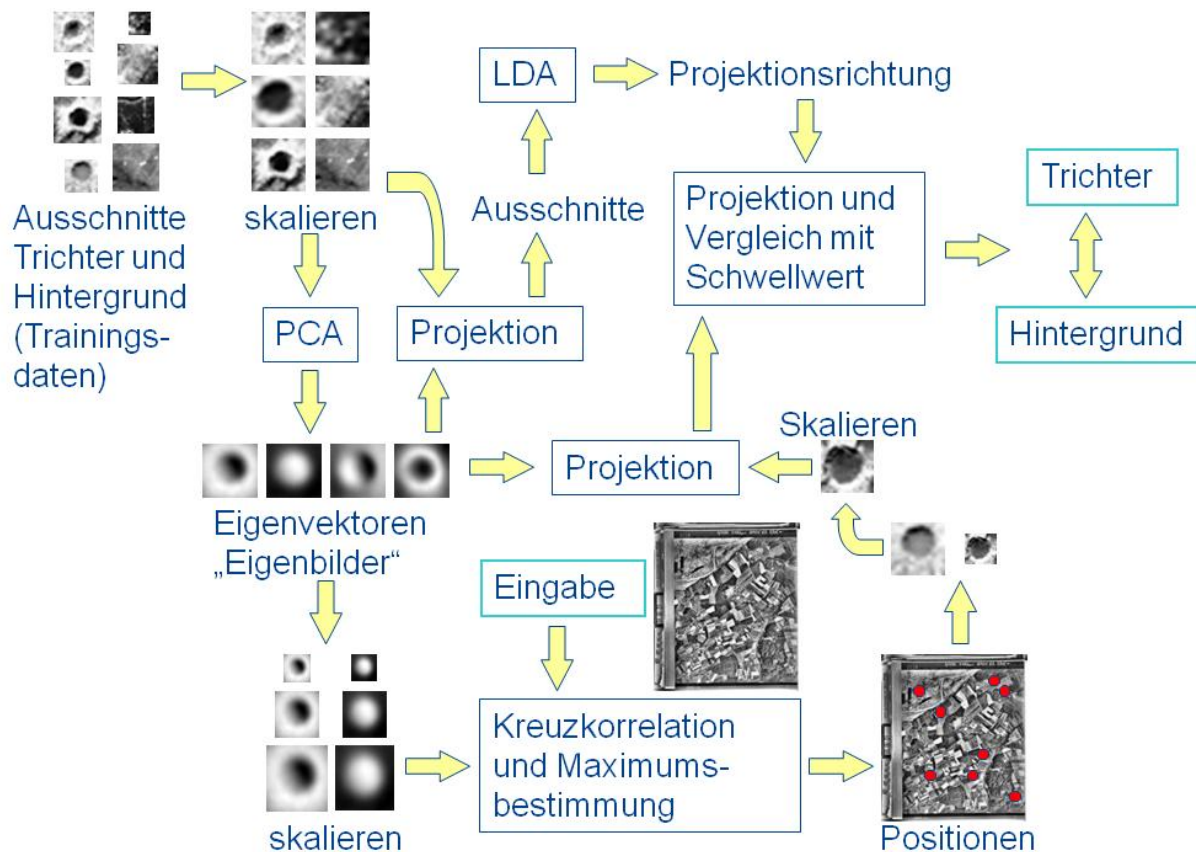


Abbildung 4: Geplanter Ablauf des Programms zur Detektion von Bombentrichtern in Luftbildern

Ausschnitten wird eine lineare Diskriminanzanalyse nach Fisher<sup>4</sup> durchgeführt, die dazu dient, die Richtung  $w$  zu bestimmen, die die Daten so in den 1-dimensionalen Raum projiziert, dass sie dort optimal nach ihrer Klasse getrennt werden können. Das Verfahren zur Bestimmung von  $w$  wird in Abschnitt 2.2.2 beschrieben.

Der Trainingsteil ist nun abgeschlossen, es folgt der Teil der Trichterdetektion: Auf einem Eingabebild werden  $K \cdot M$  Kreuzkorrelationen mit den  $M$  „Eigenbildern“ in  $K$  Größen als Templates durchgeführt und die Korrelationsmaxima bestimmt. So erhalten wir mögliche Positionen für Bombentrichter im Bild. Um diese Positionen herum werden Bildfenster ausgeschnitten, auf die Größe der Trainingsausschnitte skaliert, und ebenso wie die Trainingsausschnitte mit den  $M$  Eigenvektoren der Hauptkomponentenanalyse in einen  $M$ -dimensionalen Raum projiziert. Danach können sie mit dem im Training bestimmten Vektor  $w$  in den 1-dimensionalen Raum projiziert werden und mit Hilfe eines Schwellwertes als Trichter oder Hintergrund klassifiziert werden.

## 2.2 Methoden

### 2.2.1 Hauptkomponentenanalyse

Das Ziel bei der Hauptkomponentenanalyse ist, eine Menge von  $N$  Punkten aus dem  $D$ -dimensionalen Raum so in einen  $M$ -dimensionalen Raum ( $M < D$ ) zu projizieren, dass möglichst wenig Information verloren geht. So kann man ursprünglich hochdimensionale Daten auf ihre wesentlichen

<sup>4</sup>engl.: Linear Discriminant Analysis (LDA)

Merkmale reduzieren. Die Idee ist nun, eine Hauptachsentransformation auf den Daten durchzuführen und sie in den von  $M$  Hauptachsen aufgespannten Raum zu projizieren. Abbildung 5 zeigt ein Beispiel für eine 2-dimensionale Punktwolke, die mit der ersten Hauptachse in den 1-dimensionalen Raum projiziert wird.

Gehen wir zunächst von einer Projektion der  $x_n$  ( $n = 1..N$ ) Punkte in den 1-dimensionalen

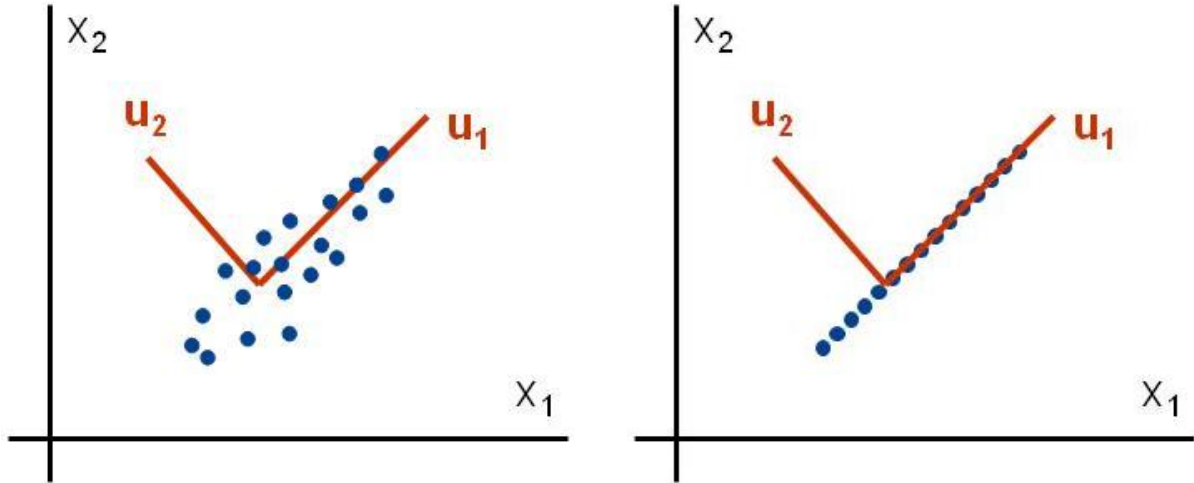


Abbildung 5: links: 2D-Punktwolke mit Hauptachsen - rechts: Projektion auf die erste Hauptachse

Raum ( $M = 1$ ) aus. Die Gleichung zur Berechnung des projizierten Punktes  $y_n$  lautet dann:

$$y_n = \mathbf{u}_1^T \mathbf{x}_n$$

Hierbei ist der Vektor  $\mathbf{u}_1$  so zu bestimmen, dass die Varianz  $\sigma_y^2$  der projizierten Daten maximal wird. Dies gewährleistet einen minimalen Informationsverlust bei der Projektion.  $\sigma_y^2$  berechnet sich nach dem Varianzfortpflanzungsgesetz aus der Kovarianzmatrix  $\mathbf{S}$  der Daten  $x_n$  und dem Vektor  $\mathbf{u}_1$ :

$$\sigma_y^2 = \mathbf{u}_1^T \mathbf{S} \mathbf{u}_1$$

Zur Berechnung benötigen wir also den Mittelwert  $\bar{\mathbf{x}}$  und die Kovarianzmatrix  $\mathbf{S}$  der Punkte  $x_n$ :

$$\bar{\mathbf{x}} = \frac{1}{N} \sum_{n=1}^N \mathbf{x}_n$$

$$\mathbf{S} = \frac{1}{N} \sum_{n=1}^N (\mathbf{x}_n - \bar{\mathbf{x}})(\mathbf{x}_n - \bar{\mathbf{x}})^T$$

Die sich die Varianz  $\sigma_y^2$  mit der Länge von  $\mathbf{u}_1$  vergrößert und somit beliebig groß werden kann, führen wir eine Nebenbedingung ein:

$$\mathbf{u}_1^T \mathbf{u}_1 = 1$$

Zu bezüglich  $\mathbf{u}_1$  zu maximierende Funktion lautet also:

$$\mathbf{u}_1^T \mathbf{S} \mathbf{u}_1 + \lambda_1 (1 - \mathbf{u}_1^T \mathbf{u}_1)$$

Nach Nullsetzen der ersten Ableitung ergibt sich die folgende Gleichung:

$$\mathbf{S}\mathbf{u}_1 = \lambda_1\mathbf{u}_1$$

$\mathbf{u}_1$  muss also ein Eigenvektor von  $\mathbf{S}$  sein. Multipliziert man die Gleichung von links mit  $\mathbf{u}_1^T$  und nutzt die Nebenbedingung aus, so erhält man für die maximale Varianz  $\sigma_y^2$  der Daten den Wert  $\lambda_1$ .

$$\mathbf{u}_1^T \mathbf{S} \mathbf{u}_1 = \lambda_1 = \sigma_y^2$$

Offenbar wird  $\sigma_y^2$  dann maximal, wenn  $\mathbf{u}_1$  als der zum größten Eigenwert  $\lambda_1$  gehörige Eigenvektor der Kovarianzmatrix  $\mathbf{S}$  gewählt wird.  $\mathbf{u}_1$  wird als erste Hauptkomponente bezeichnet.

Für die Erweiterung auf die Projektion in den  $M$ -dimensionalen Raum wählen wir als nächste Hauptkomponente den Vektor  $\mathbf{u}_2$ , der die projizierte Varianz orthogonal zu  $\mathbf{u}_1$  maximiert. So verfahren wir für die weiteren  $\mathbf{u}_m$  ( $m = 3..M$ ) Vektoren. Als  $M$  Hauptkomponenten erhalten wir damit die zu den  $M$  größten Eigenwerten  $\lambda_1.. \lambda_M$  gehörenden Eigenvektoren  $\mathbf{u}_1.. \mathbf{u}_M$ . Die Projektion der Daten in den  $M$ -dimensionalen Unterraum ergibt sich dann mit den in einer Matrix  $\mathbf{U}$  zusammengefassten  $M$  Eigenvektoren nach der Formel

$$\mathbf{y}_n = \mathbf{U}^T \mathbf{x}_n$$

$$\mathbf{U} = \begin{bmatrix} \mathbf{u}_1 & \mathbf{u}_2 & \cdots & \mathbf{u}_M \end{bmatrix}$$

### 2.2.2 Fisher's lineare Diskriminanzanalyse

Gegeben sind  $N_1$  Punkte in der Klasse  $\mathcal{C}_1$  und  $N_2$  Punkte in der Klasse  $\mathcal{C}_2$  im  $D$ -dimensionalen Raum. Bei Fisher's linearer Diskriminanzanalyse ist ein Vektor  $\mathbf{w}$  gesucht, der die Punkte so in den 1-dimensionalen Raum projiziert, dass die beiden Klassen dort möglichst gut trennbar sind. Abbildung 6 stellt die Situation dar.

Die Idee ist,  $\mathbf{w}$  so zu wählen, dass die Varianz zwischen den Klassen, also der Abstand der

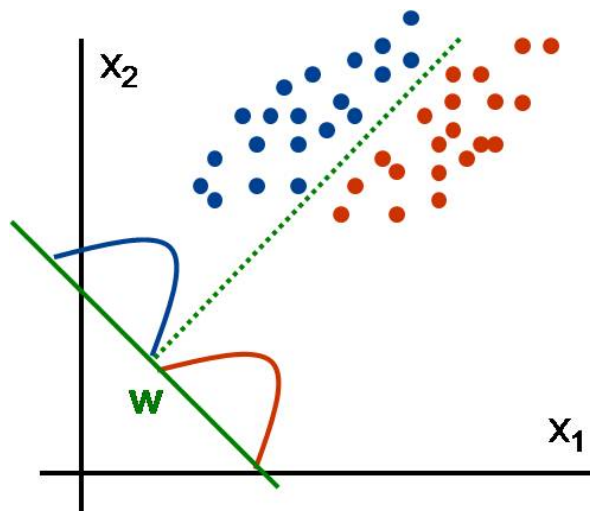


Abbildung 6: blau: Punkte der Klasse  $\mathcal{C}_1$  - rot: Punkte der Klasse  $\mathcal{C}_2$

projizierten Klassenmittelpunkte, möglichst groß wird. Dabei soll die Varianz der projizierten Punkte innerhalb der einzelnen Klassen aber möglichst klein bleiben (siehe Abbildung 7).

Man kann zeigen, dass dies erfüllt ist, wenn der sog. Rayleigh-Quotienten  $J(\mathbf{w})$  bezüglich  $\mathbf{w}$

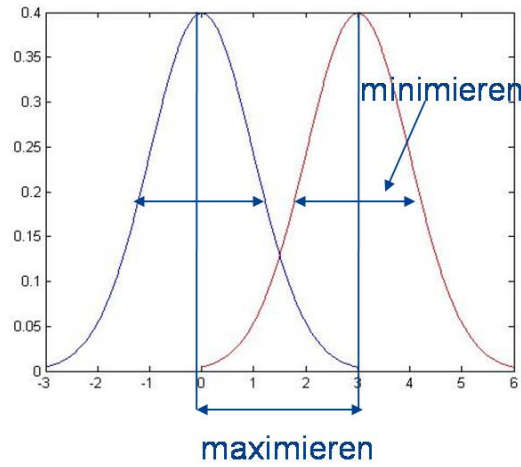


Abbildung 7: blau: Verteilung der Punkte in Klasse  $\mathcal{C}_1$  - rot: Verteilung der Punkte in Klasse  $\mathcal{C}_2$

maximiert wird.

$$J(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{S}_B \mathbf{w}}{\mathbf{w}^T \mathbf{S}_W \mathbf{w}}$$

$\mathbf{S}_B$  ist dabei die Kovarianzmatrix zwischen den Klassen

$$\mathbf{S}_B = (\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^T$$

und  $\mathbf{S}_W$  die Kovarianzmatrix innerhalb der Klassen.

$$\mathbf{S}_W = \sum_{n \in \mathcal{C}_1} (\mathbf{x}_n - \mathbf{m}_1)(\mathbf{x}_n - \mathbf{m}_1)^T + \sum_{n \in \mathcal{C}_2} (\mathbf{x}_n - \mathbf{m}_2)(\mathbf{x}_n - \mathbf{m}_2)^T$$

$\mathbf{m}_1$  und  $\mathbf{m}_2$  sind die Klassenmittelpunkte.

$$\mathbf{m}_1 = \frac{1}{N_1} \sum_{n \in \mathcal{C}_1} \mathbf{x}_n \quad \mathbf{m}_2 = \frac{1}{N_2} \sum_{n \in \mathcal{C}_2} \mathbf{x}_n$$

Maximiert man den Rayleigh-Quotienten, ergibt sich als Lösung für  $\mathbf{w}$ :

$$\mathbf{w} = \mathbf{S}_W^{-1}(\mathbf{m}_2 - \mathbf{m}_1)$$

Fisher's lineare Diskriminanzanalyse wurde hier nur sehr knapp erläutert, in der Bachelorarbeit wird die Herleitung ausführlicher dargestellt werden.

### 3 Bisherige Umsetzung

Das Programm soll in Matlab implementiert werden. Dabei wird die an der technischen Universität in Prag entwickelte STPR-Toolbox<sup>5</sup> für Matlab verwendet. Dort sind u.A. die Hauptkomponentenanalyse und Fisher's lineare Diskriminanzanalyse implementiert. Da es aufgrund der Größe der Bilder bereits beim Einlesen Speicherplatzprobleme gab, wurden sie von ca. 11000 x 12000 Pixel auf ca. 2000 x 2300 Pixel verkleinert.

Die zur Verfügung stehenden Bilder wurden in drei Kategorien eingeteilt:

- Trainingsmenge: bestehend aus 24 Luftbildern (mit und ohne Trichter) aus dem ländlichen Raum, die zur Erstellung der Templates und der Projektionsrichtung für die Klassifikation dienen.
- Testmenge 1: bestehend aus 30 Luftbildern (mit und ohne Trichter) aus dem ländlichen Raum, die zum Testen des Algorithmus mit Bildern, die denen der Trainingsmenge ähneln, dienen.
- Testmenge 2: bestehend aus 20 Luftbildern aus der Stadt, mit extrem vielen Trichtern oder von schlechter Qualität, die zum Testen des Algorithmus mit Bildern, die sich stark von denen der Trainingsmenge unterscheiden, dienen.

Die Anwendung des Algorithmus auf die Testmenge 2 soll nur dazu dienen, die Leistungsfähigkeit auf Bildern, die nicht den Voraussetzungen entsprechen, festzustellen. Der Algorithmus soll im Rahmen dieser Bachelorarbeit nicht explizit für derartige Bilder erstellt oder angepasst werden. Die Durchführung der Hauptkomponentenanalyse mit ca. 4000 Bildausschnitten der Größe 31 x 31 Pixel, bei einem Verhältnis von 1:3 von Trichtern zu Hintergrund, lieferte das in Abbildung 8 dargestellte Ergebnis für die ersten 20 „Eigenbilder“. Man erkennt in den meisten Bildern deutlich eine runde Form, die an Bombentrichter mit eventuellem sichelförmigen Schattenwurf erinnert. Somit scheinen die „Eigenbilder“ für die Nutzung als Templates für die Kreuzkorrelation geeignet.

Die ca. 4000 Bildausschnitte aus 10 Bildern wurden mit den ersten 20 Eigenvektoren in den 20-dimensionalen Raum projiziert und, wie in Abschnitt 2.1 beschrieben, eine lineare Diskriminanzanalyse durchgeführt. Die zweidimensionale Darstellung der Daten liefert das Ergebnis in Abbildung 9. Die abgebildete Gerade stellt die Trennfunktion zwischen beiden Klassen dar. Man erkennt, dass sich die Klassen teilweise überlappen, und somit keine fehlerfreie Klassifikation mit der LDA zu erzielen ist. Die Fehlerrate scheint allerdings so klein zu sein, dass die Verwendung einer solchen Trennfunktion gute Klassifikationsergebnisse für die Testdaten erwarten lässt. Insgesamt lassen die bisherigen Ergebnisse darauf schließen, dass das gewählte Konzept ein plausibler Ansatz zur Detektion der Bombentrichter ist.

### 4 Weiteres Vorgehen

Bei der weiteren Umsetzung des Verfahrens sollen zunächst die PCA und die LDA mit allen Ausschnitten aus den 24 Trainingsbildern durchgeführt werden. Der nächste Schritt ist die

---

<sup>5</sup>STPR - Statistical Pattern Recognition



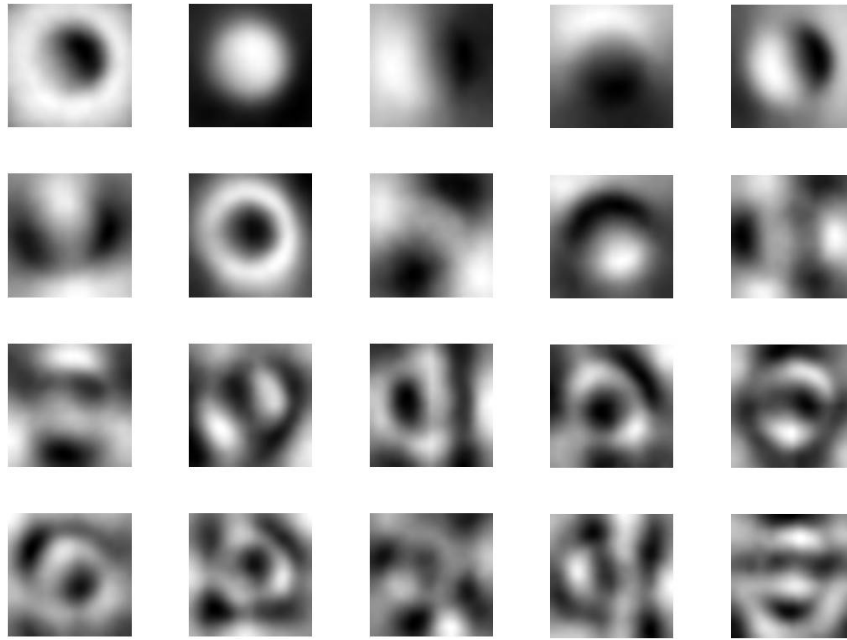


Abbildung 8: Die ersten 20 „Eigenbilder“ der Hauptkomponentenanalyse mit ca. 4000 Bildausschnitten

Durchführung der Kreuzkorrelation mit den erstellten Templates auf den Testbildern, wobei u.A. ein Schwellwert für die Maximumsbestimmung gefunden werden muss. Des Weiteren muss eine Trennfunktion für die Klassifikation ermittelt und die Fenster an den potentiellen Trichterpositionen in den Testbildern klassifiziert werden. Der Algorithmus soll getestet und daraufhin durch Verändern der Parameter optimiert werden. Die wichtigsten Parameter, die für das Verfahren optimal bestimmt werden müssen, werden hier mit den in diesem Zusammenhang zu klärenden Fragen stichwortartig aufgelistet:

- die Anzahl  $M$  der „Eigenbilder“. Wie viele sind nötig, um möglichst viele Trichter zu finden?
- die Ausschnittsgröße. Auf welche Größe werden die Ausschnitte skaliert? Hat dies überhaupt einen Einfluss auf die Form der Eigentrichter und/oder die Klassifikation? Wie viel von der Umgebung der Trichter wird mit ausgeschnitten?
- die Anzahl der Skalen. In wie vielen und welchen Größen werden die Templates mit dem Bild korreliert um alle Trichter zu erfassen?

## 5 Zeitplan

Für die mit 360 Stunden (inklusive Oberseminar) angesetzte Bachelorarbeit ist der in Abbildung 10 dargestellte Zeitplan vorgesehen.

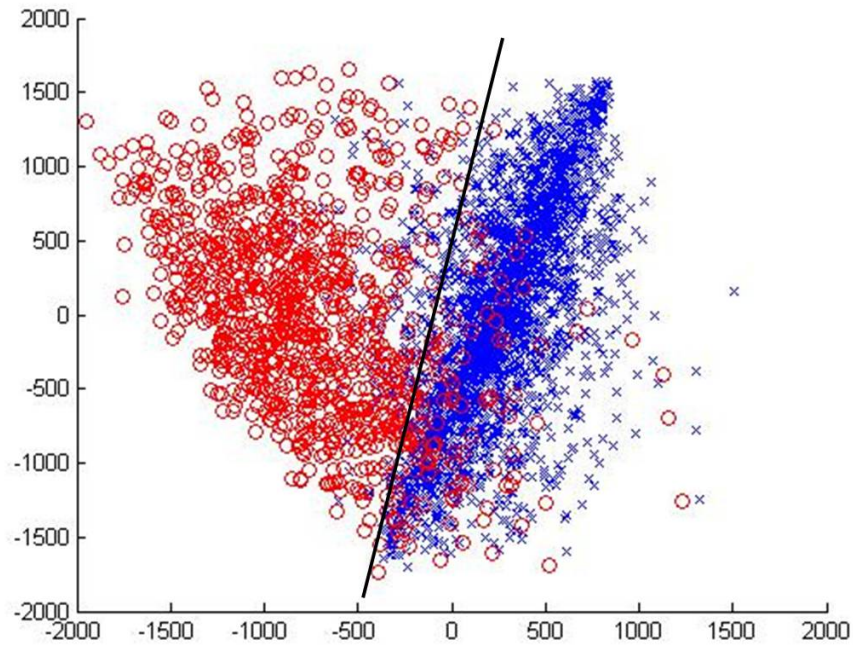


Abbildung 9: Zweidimensionale Darstellung der Bildausschnitte nach der LDA und der Trennfunktion, rot: Trichter - blau: Hintergrund

## 6 Quellen

*Bishop, Christopher M.* 2006. Pattern Recognition and Machine Learning.

*Förstner, Wolfgang.* Skript zur Vorlesung Bildinterpretation 2007/08

## Zeitplan Bachelorarbeit

KW

8	<b>Block Theorie &amp; Konzept</b>	
9	<b>sowie Oberseminar (110 h)</b>	
10	Theorie & Konzept	60
11	Schreiben	10
12	Umsetzen	20
13	Testen	20
14	<b>Block Implementierung (170 h)</b>	
15	Umsetzen	70
16	<b>17.04. Abgabe Oberseminar</b>	
17	Testen	65
18	Theorie & Konzept	25
19	<b>IGSM</b>	
20	Schreiben	10
21	<b>23.05. Abschluss Implementierung</b>	
22	<b>Praktikum</b>	
23	<b>Block Schreiben (80 h)</b>	
24	Schreiben	70
25	Korrektur/Layout	10
26	<b>27.06. Abgabe Bachelorarbeit</b>	

### Inhaltliche zeitliche Gliederung

	bis
1 Trainings- und Testdaten zusammenstellen	04.04.
2 Erstellung der Templates und der Trennfunktion für die Klassen	24.04.
3 Kreuzkorrelation, Maximumsbestimmung und Klassifikation auf den Testbildern	02.05.
4 Empirische Tests und Optimierung	23.05.

#### Zeitaufwand in Stunden:

Theorie & Konzept	85
Umsetzen	90
Testen	85
Schreiben	100
<b>Summe</b>	<b>360</b>

Abbildung 10: Zeitplan für die Bachelorarbeit

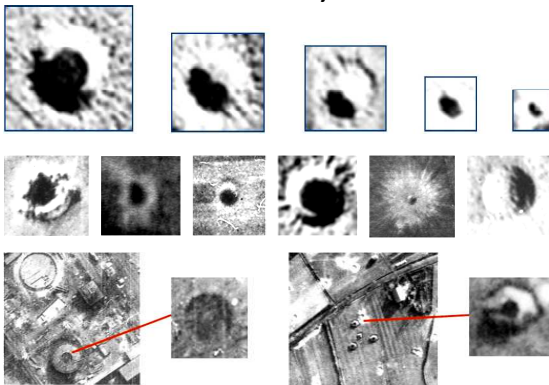
Anhang D

Plakative Darstellung

## Problemstellung

**Aufgabe** Die Suche nach Blindgängern in Luftbildern kann durch **automatische Bombentrichterdetektion** erleichtert werden.

**Anforderungen** **Gestaltvariationen** in Größe und Form müssen berücksichtigt werden. Bombentrichter ähneln auch anderen Objekten im Bild.



Bsp. für Gestaltvariationen und trichterähnliche Objekte

## Konzept und Umsetzung

Bombentrichterdetektion in zwei Schritten:

**Kandidaten-suche** Durch **Kreuzkorrelation** des Bildes mit einem repräsentativen Trichter-Template in verschiedenen Größen.

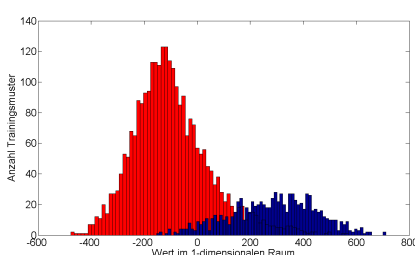
**Klassifikation der Kandidaten** Merkmale sind die Grauwerte der Bildausschnitte. Klassifizierung anhand der **Wahrscheinlichkeitsdichte** der Klassen Trichter und Hintergrund.

Zur Bestimmung der Verteilungsparameter ist eine **Dimensionsreduktion** des Merkmalsraums notwendig



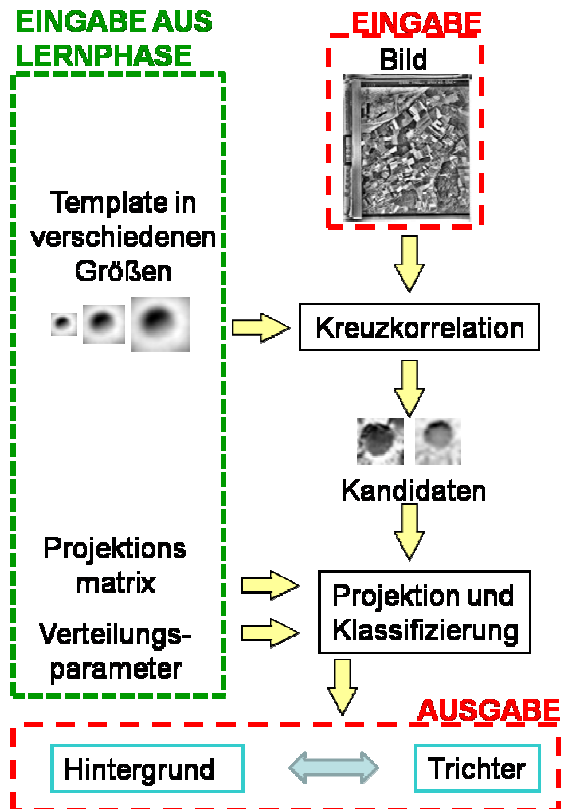
Projektion der Ausschnitte in einen Unterraum mittels aus **PCA und LDA** auf Trainingsmustern bestimmter Projektionsmatrizen

**Umsetzung** Projektion der Trainingsmuster in 1D, Annahme einer Normalverteilung



Verteilung der Trainingsmuster in 1D. (Rot: Hintergrund, Blau: Trichter)

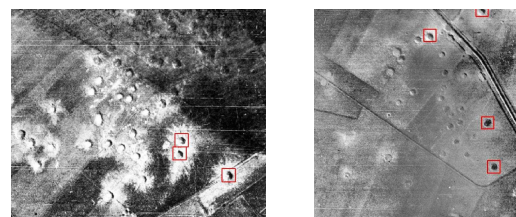
## Ablauf der Bombentrichterdetektion



## Ergebnisse

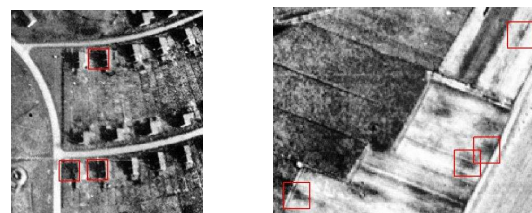
• Zwischen **64% und 70%** der Bombentrichter auf den Trainingsbildern werden detektiert.

Beispiele für **false negatives**:



• Die Relevanz liegt bei durchschnittlich 31%, abhängig von der Dichte der Bombardierung.

Beispiele für **false positives**:



• Ableitung von Vorschlägen für weitere Trichterklassen

Vielversprechende Ergebnisse bei der Einordnung von Testbildern in **Bombentrichter-Dichte-Kategorien**